CONTRACT REPORT ARBRL-CR-00457

AFATL-TR-81-60

IMPROVEMENTS IN EPIC3-LINK TO HULL, IMPROVED EQUATION-OF-STATE, IMPROVED FRACTURE MODELLING, SAIL UPDATE SYSTEM, OTHER IMPROVEMENTS

Prepared by

Orlando Technology, Incorporated
P. O. Box 855
Shalimar, Florida 32579

June 1981

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CONTRACT REPORT ARBRL-CR-00457	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* Improvements in EPIC3-Link to HULL, Improved Equation-of-State, Improved Fracture Modelling, SAIL Update System, Other Improvements		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) John J. Osborn		8. CONTRACT OR GRANT NUMBER(s) DAAK11-79-C-0106
9. PERFORMING ORGANIZATION NAME AND ADDRESS Orlando Technology, Incorporated P. O. Box 855 Shalimar, Florida 32579		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Armament Research and Development Command US Army Ballistic Research Laboratory ATTN: DRDAR-BL Aberdeen Proving Ground, MD 21005		12. REPORT DATE JUNE 1981 13. NUMBER OF PAGES 87
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

AFATL-TR-81-60

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Lagrangian	Impact
Hydrocode	Penetration
Eulerian link	

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report documents research performed to enable the EPIC 3D Lagrangian hydrocode to run in a linked mode with the HULL 3D Eulerian hydrocode. The linked mode is required to efficiently and accurately model multiple target plate impact and penetration events. The report also documents many changes made to make EPIC more user oriented and simple to run.

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

FOREWORD

This report documents EPIC3 improvements developed under BRL contract DAAK11-79-C-0106. The work was performed by John J. Osborn, Orlando Technology, Inc. during the period July 1979 through July 1980. The improvements allow EPIC to be run more easily, to more realistically handle material fracture and to be linked with HULL for multiple penetration problems. The BRL project manager was Dr. John Zukas. Thanks are due Dr. Zukas and Mr. Kent Kimsey, of this organization, for their support and assistance during this effort.

ACKNOWLEDGEMENT

EPIC3 is a Lagrangian three-dimensional wave propagation computer code developed for the BRL by Dr. Gordon Johnson of Honeywell. The work performed by the author under this contract built upon Dr. Johnson's work and primarily consists of programming modifications using the basic EPIC structure.

Many of the programming changes would not have been possible without the prior development of SAIL and HULL library routines by government and contractor personnel too numerous to mention by name.

The link philosophy and implementation built heavily upon previous work in two dimensions supported by the Air Force Armament Laboratory.

TABLE OF CONTENTS

LIST OF FIGURES

SECTION I

INTRODUCTION

The primary purpose of this development contract was an EPIC link with HULL to allow the Ballistics Research Laboratory to run multiplate penetration problems with relative ease. The link is patterned after a similar link between HULL and TOODY for two-dimensional problems.

In addition to the linking routines, a great deal of time was spent in making EPIC simpler to run in a production mode. Several EPIC routines were combined under the SAIL update system. As a result EPIC routines are automatically sized to use the minimum core possible. Sizing includes array sizing as well as insertion or deletion of large blocks of coding. EPIC time history station routines were modified to allow the user to input derived positions (instead of node and element numbers), to save data based on a percentage change in key variables since the last save, and to pack the data into optimum size buffer areas. The time history plotter was modified to allow any amount of data to be plotted by deleting the requirement that the data for a station fit in memory.

In addition, a more realistic metal equation of state was included in EPIC along with a failure model based on principle tensile strain.

Numerous smaller programming changes were made - such as format free input, tape buffer size reductions, debug points if desired, etc.

These programming changes, along with details of the link are described in this report.

SECTION II

HULL/EPIC LINK

Lagrangian wave propagation codes such as EPIC have great difficulty completing a plate penetration calculation due to the large distortions which occur as the target crushes, strains to very large values and petals or plugs open. It is usually impossible to compute such problems correctly or to maintain a reasonable time step. However, this is the type of violent flow for which Eulerian codes were developed. They can handle large deformations with ease.

On the other hand, in a multiplate target with long separation distances between plates, an Eulerian mesh would have to be extremely finely zoned to properly calculate stresses and strains in the slowly deforming projectile as it passes between plates. If the plates were separated by any appreciable distance, the long flight time between plates would make the Eulerian calculation prohibitively expensive. A Lagrangian code would excel in this region since the deformations set up by first plate impact are relatively mild. And the Lagrangian code would handle this phase efficiently since only the projectile need be in the problem.

The most efficient multiplate impact calculation would therefore be one which uses the type of code most appropriate in each time regime. This is the basic rationale behind the development of the HULL/EPIC link.

HULL will handle the first plate penetration and then, in effect, hand the problem over to EPIC for the long flight to the next plate. Upon reaching this next plate, EPIC can hand the problem back to HULL for the plate penetration. If there is penetrator left after this second plate penetration, the problem can be returned to EPIC, then returned to HULL for the next plate, etc.

Actual implementation of the link proceeds as follows:

1. A HULL problem is established which models the first target plate and some or all of the projectile. Time history stations are located in the nose region of the projectile. The HULL problem is then run until the first plate is penetrated and plate influence on the projectile ceases or reaches insignificant levels.

2. An EPIC calculation is then set up which describes the projectile prior to first plate impact. The target is not modelled in the EPIC calculation. The calculation is begun and EPIC surface nodes are driven by velocities from appropriate time

history station data stored on the tape made during the HULL run. Velocities are used instead of stresses since the EPIC boundary must be driven with displacement or one of its derivatives. Stress in the target next to the projectile could be used but would require a complicated routine applying that stress to the appropriate element surface. After the EPIC calculation passes through the time regime for which HULL time history station data exists, the nodes which were driven by this data automatically revert to free surface nodes and the calculation continues toward the next plate. Just prior to reaching the point in space where the second plate would exist, the EPIC calculation is terminated with a normal dump which contains all EPIC element and node information at termination time.

3. The EPIC dump is used as input to a new HULL calculation which will describe the projectile and the next target plate.

4. The EPIC problem can be continued through the second plate and beyond by using second HULL run station data to drive EPIC nodes beginning at the time of the last EPIC dump.

5. Upon reaching the third plate, the second EPIC run can be terminated and the data transferred back into HULL for a third plate impact.

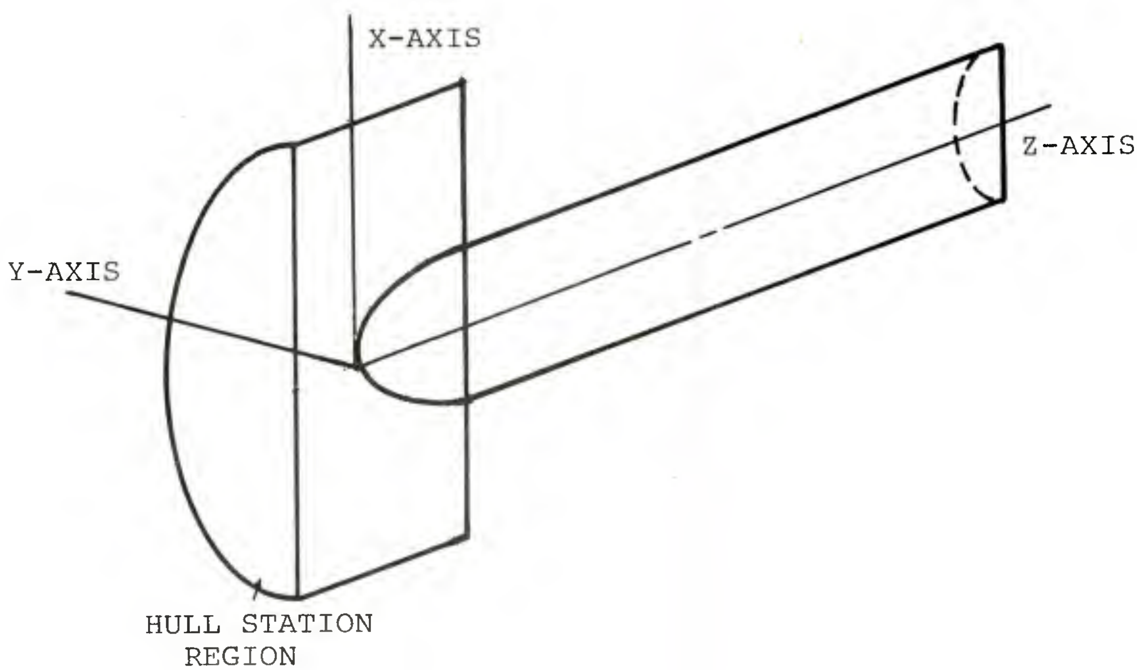
6. This process can continue indefinitely as long as some of the penetrator survives.

The amount of projectile which has to be modelled in the initial HULL run depends upon the material properties of the projectile and the time at which significant plate influence on projectile motion ceases. Enough of the projectile should be modelled to insure that velocity data in the HULL time history stations is valid. It would be undesirable to have signals return from the aft end of the projectile to its nose during the HULL run, unless such signals would exist in the fully modelled projectile. The time of plate influence will change as projectile velocity changes, its obliquity changes or the thickness of the plate changes. It is not possible therefore to determine a single time of influence to apply to all problems. The time of influence can be as little as 20 to 25 microseconds for a projectile impacting a 3/8 inch RHA plate at 2 km/sec or as long as 100 to 120 microseconds for the same projectile slowed to 1 km/sec at an obliquity angle of 65 degrees. Whatever the specific situation the length of projectile modelled should be the real length or one half the time of influence multiplied by the projectiles' elastic wave speed, whichever is smaller. For example, if the normally impacting 2 km/sec projectile is copper which has an elastic wave speed of 5 km/sec the length required to be modelled is approximately 6 centimeters - unless, of course, this exceeds the real length of the projectile. If the obliquily impacting 1 km/sec projectile is staballoy with

an elastic wave speed of 3.2 km/sec, we must model almost 20 centimeters of length to insure valid HULL station data.

It would be highly desirable to not require EPIC to "repeat" the HULL phase. In theory, this could be accomplished by extracting projectile information from a HULL dump made at the time plate influence ceases. In practice this requires a three-dimensional finite element generator capable of providing reasonable zoning in a completely arbitrary region. It would also require that the projectile in HULL be very finely zoned so that even low level stresses are accurately defined at transfer time. The former requirement would be very difficult to achieve without quite a bit of manual and time consuming intervention. The latter requirement means that whatever efficiency were gained by not repeating the EPIC calculation would probably be lost in the extra calculational time required for a very finely zoned HULL calculation. The best approach to avoid "repeating" the plate impact and to insure adequate accuracy would appear to be a fully interactively linked HULL/EPIC system.

The most accurate HULL initial penetration setup is one in which the penetrator is oriented along the Z-axis. In this case, time history stations should be used which move in a Lagrangian fashion in the Z direction and are Eulerian (i.e., fixed in space) in the other directions. The difficulty in using purely Lagrangian stations is simply that such stations do not exactly follow the flow and they can move so far as to become useless. Stations Lagrangian in the Z direction should be uniformly placed within the projectile's nose region through the projectile and radially to distances of two to three times the projectile's radius as illustrated schematically below.



This will insure that stations always exist in the projectile material and near the projectile's nose surface. Of course, if significant target impact is expected at other locations along the projectile, HULL stations should be located at those points also as might occur in a projectile with a highly flared aft body section. The HULL station tape is packed to reduce the length required and station data is not collected for a station if it has not changed by more than 2 percent from the last time it was saved. Therefore, there is no overriding reason to attempt to severely limit the number of HULL stations. A few hundred HULL stations can be used without causing any calculational difficulties. Even more stations could be used by reducing the amount of information carried at each station. Data carried in each station for a three-dimensional run is:

Density (with station number masked into the lower bits of the word)

X position (with material data masked into the lower bits of the word)

X velocity

Y velocity

Z velocity

X acceleration

Y acceleration

Z acceleration

$$\left.\begin{array}{l}\tau_{XX}\\ \tau_{YY}\\ \tau_{ZZ}\\ \tau_{XY}\\ \tau_{XZ}\\ \tau_{YZ}\end{array}\right\}\ \text{TOTAL STRESS COMPONENTS}$$

$$\left.\begin{array}{l}\varepsilon_{XX}\\ \varepsilon_{YY}\\ \varepsilon_{ZZ}\\ \varepsilon_{XY}\\ \varepsilon_{XZ}\\ \varepsilon_{YZ}\end{array}\right\}\ \text{TOTAL STRAIN COMPONENTS}$$

Y Position

Z Position

I Internal energy/unit mass (if NVARST=23)

The Hull variable NVARST - contained in the HULL Z block - indicates the number of variables carried per station. In the normal HULL run this is set to 22. The user can reset it to 23 and HULL will automatically also carry internal energy per unit mass. It is a relatively simple matter to change HULL to carry fewer variables - just the station positions and velocities for instance - and to change EPIC to accept this reduced amount of information. NVARST could be changed to a smaller number in KEEL input and coding deleted or changed in HULL subroutine STATON which causes variables to be saved in the STAD array. The EPIC changes are even simpler. In EPIC subroutine RDH, one need change only the arguments in the buffered in AHULL array in loops in which this information is transferred to the EPIC variables XHULL, YHULL, ZHULL, UHULL, VHULL and WHULL. These latter arrays contain HULL station X, Y, Z, U, V and W values.

The material information placed in the lower bits of the X position is binary information which describes whether HULL materials are in the zone (1) or not in the zone (0) at the time of station updating. For example, if HULL is running with three materials (CU, RHA and AIR) and the HULL material numbers are:

CU = 1
RHA = 2
AIR = 3

Then if a zone contains only air the lower bits in the density word will be:

100

If the zone contains air and copper the bits will be:

101

If all three materials are contained in the zone, the bits are:

111

As seen, the material data is stored from right to left in bits which correspond to the material number. When EPIC examines a HULL station, it unmasks the material type word and decides if the word indicates that HULL projectile material was contained in the zone at the time the station was updated. The HULL material numbers for projectile materials are carried in the MATHUL array which is input to EPIC at the time the EPIC MAIN program is run. Because the HULL material type word contains 15

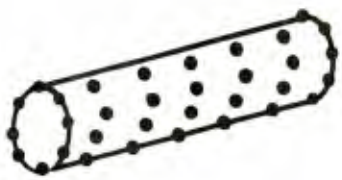
binary bits, the number of materials identified in the word is limited to 15. In the case illustrated, if the projectile is copper then MATHUL would be 1. The material type word is logically added (AND) to $2^{(MATHUL-1)}$. If the result is greater than zero projectile material exists in the station and the station can be used to drive EPIC surface nodes. (Recall that in an "AND" operation, 1 and 0 become 0, 0 and 1 become 0, 0 and 0 remain 0. Only 1 and 1 will become 1 - i.e., a number greater than zero). In a linked run the ISLAVE node variable is used to contain EPIC material number for the associated element. The EPIC MAIN program not only asks for a MATHUL array, but also for a corresponding MATEPIC array. The arrays indicate the relationship between HULL materials and EPIC materials. An EPIC node is driven only by HULL stations which contain some of the appropriate material. The relationship in this array can be changed after the second and subsequent HULL runs.

HULL station data is contained in two arrays. One array contains data valid at some HULL time, t_1, and one array contains the data at the next HULL time, t_2, where EPIC time is equal to one of these limits or between them.

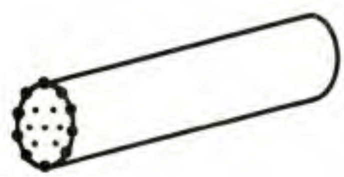
When EPIC requires HULL velocity for a node it searches through both arrays for all stations which contain the appropriate material and which are within a DELTA distance from the node. DELTA is set to 1 centimeter but can be changed to any value by changing the data statement in MAIN subroutine HVEL. Note that for a linked run, EPIC must use the CGS system of units. However, for an unlinked EPIC voyage under SAIL, any consistent set of units is permitted. The station nearest to the node is used if it contains the correct material at both times. There is no spatial interpolation employed. Once the station has been found, velocity components are established for the node by linearly interpolating between data at t_1 and t_2 to find data valid at the current EPIC time. This HULL velocity data is then used directly to drive the node in question. If a station cannot be found, the X-component of velocity is returned as -7777777 and velocity is computed for the node as would normally be accomplished using the forces on the elements surrounding the node. When HULL data ceases, the variable NHULL is set to zero and node velocities are calculated solely from forces on the elements.

NHULL was set to 1 by the EPIC generator when the user asked that surface nodes for certain projectile regions be identified as being driven by HULL input. Nodes on the surfaces of rods and nose sections can be so identified. Two generator input variables are used for this purpose for a rod. If NHSURF is set to 1, the generator assumes that all cylindrical surface nodes will be driven by HULL - these are all front surface nodes including those on the cylindrical surface. If both NHFRNT and NHSURF are

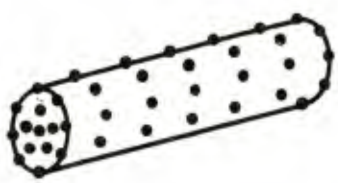
1 all surface nodes - except those on the aft face - are driven by HULL input. These regions are illustrated below where dots identify nodes driven by HULL input.



NHSURF=1



NHFRNT=1



NHSURF=1
and
NHFRNT=1

For nose sections there is only one input variable - NHSURF. If it is 1 all nose section surface nodes are driven by HULL input.

Nodes to be driven by HULL stations are identified with a 1 in the ten thousandths digit in the constraint word for the node.

It is somewhat computationally faster to identify nodes as driven by HULL only if they can reasonably be expected to come in contact with the target plate and only, of course, if they are in or near a region covered by HULL time history stations. The DELTA check prevents nodes from being driven by HULL if they are too far removed from HULL stations, but some computer time will be lost in the station search which is conducted every cycle for every node identified as being driven by HULL.

The projectile nose section should always be driven by HULL input. Portions of the rod aft of the nose section may or may not be so driven - depending on the particular problem.

Of course, non-surface nodes should never be driven by HULL. If a nose section exists on the projectile then NHFRNT should be set to zero for any attached rod sections.

It is always desirable to include the complete projectile in the EPIC calculation. However, in some impact situations - soft metals at high velocities - we can expect the nose section to become highly deformed. This deformation can drive the EPIC time step to values too small to reasonably continue the problem. If this occurs, the nose section and even part of the rod can be dropped from the EPIC grid and the problem continued using HULL stations further back in the rod. Or, the problem could begin with the nose section and some rod missing. In this case, NHFRNT should be set to 1 so that rod front nodes will be driven by HULL. Since the region which now constitutes the front of the EPIC projectile is interior to the HULL projectile the EPIC calculation will be an exact HULL duplicate until the time that the HULL stations are dropped. At this time the EPIC calculation will be less exact since the EPIC projectile is missing some mass. However, the accuracy of the EPIC run may be sufficient since much of this mass probably suffered large deformations and separated from the projectile anyway. How much nose section can be dropped from the EPIC calculation to increase its efficiency will clearly be a function of the particular penetration event being examined and the projectile data desired.

Nodes can be dropped from the projectile as the EPIC calculation is running through use of the material property EBS, by dropping projectile surface elements when they drive the time step below a given level or by dropping all elements beyond a given element number. When using a HULL metal equation of state in EPIC, maximum principle tensile strain in an element is compared to EBS. If it exceeds EBS, the element variable ICHECK is set to -1. There will then be no further calculations performed for this element. At the time the element is "dropped" from the calculation all nodes defining the element are checked to see if any are driven by HULL input. If any are so driven, then the remaining nodes - which will become surface nodes when the element is dropped - are modified so that they will be driven by HULL input. However, use of EBS requires some experience with the particular class of problems being run. Indiscriminate use can cause too much of the projectile to be dropped. We recommend that problems be run without the EBS check if possible (setting EBS to 999 in input). If this is not possible, EBS should be set to as large a value as practical for concluding the calculation. There is no input provision for changing EBS after the projectile has been generated. However, it can be changed with a simple one command insertion in subroutine RECALL after the material property record has been read.

EPIC MAIN asks for values for the variables NHELE and DTHULL if the calculation is a linked one. These variables are used in deciding if surface elements should be dropped from the projectile. If a surface element has an element number greater than NHELE and it is driving the time step below DTHULL, the element is dropped (i.e., ICHECK is set to -1). In a linked run, the EPIC generator uses the ISLAVE digit in the node identification word IXYZ, to identify a surface node. An element with three such nodes is a surface element. When an element is dropped, all nodes for the element become surface nodes so that surrounding elements will then become surface elements. The NHELE and DTHULL inputs are used primarily to drop elements which are on the projectile nose and are being highly compressed and in effect, being taken from the projectile and added to the target.

If it is not desired to use this option, set NHELE to a very large number or DTHULL to zero.

In a linked run, MAIN allows the user to specify up to fifty values of THDROP and NHEDRP, where all elements with numbers NHEDRP and beyond are dropped when problem time equals THDROP. This was implemented primarily as a means of dropping entire planes of elements where NHEDRP is the first element in the plane. However, it need not be used simply for entire planes.

When any element drop option is used, associated element mass is deleted from the element's nodes and node link indicators are moved to the new exposed surface.

Some mild rezoning, such as element plane combining, would be helpful in maintaining a large EPIC time step in high velocity, soft metal projectile impacts. Such plane combining was beyond the scope of this contract and the contract schedule precluded its development. However, it could be implemented relatively simply if the need arises.

The transfer back into HULL is accomplished very easily. The EPIC dump, valid at transfer time, is converted into a form usable by HULL through the EPIC system program HULLIN. This new dump then becomes part of the input package in KEEL, the HULL generator. Details of setup of the HULL problem are discussed in HULL documentation. The tape conversion routine HULLIN allows the user to specify X, Y and Z bounds in EPIC and material numbers for transfer to HULL. That is, not all of the problem need be transferred. This ability to specify regions for transfer means that surviving portions of a projectile can be translated into material number or spatial bounds. In addition, element time history points can be transferred into HULL and converted to HULL time history points if desired and if they are

in the material number and spatial windows. Latest element time history station positions are printed so that the user can input them into HULL. Data from the EPIC time history tape are not transferred. Elements with ICHECK = -1 are not transferred.

HULLIN creates a tape (Tape 15) with the EPIC data ready for use by KEEL. This tape is not automatically cataloged. It is up to the user to save it by external catalog or tape mounting instructions.

After the second HULL run, the EPIC run can be restarted if there is any survivable projectile material left for travel to the next plate. New MATHUL and MATEPIC arrays can be input if required.

This second EPIC run can be transferred back into HULL for the next plate impact.

This process can continue as long as there is sufficient projectile material available for continued EPIC runs.

SECTION III
RUNNING EPIC UNDER SAIL

The SAIL executive language[1,2] is used to create and update EPIC system routines. SAIL is a program which resides in executable form in the HULLIB library. Source programs for SAIL are also available. SAIL is designed to run on CDC, Honeywell and IBM machines with the change of one parameter in its input file.

SAIL uses a source file containing program card images in a special format and generates program files which are ready for compilation. The programs in this source file can be modified prior to execution through commands similar to the CDC update system. In addition, user generated "options" are available for tailoring a program in many ways. Options are defined and given default values at the time the SAIL source file is generated. These default values can be changed in a number of ways when generating a file for compilation. The options can be used to define the size of common arrays, to keep or delete coding and even to be used by SAIL "if" commands to define values for other options.

For example, assume the option XSIZE has been defined at generate time i.e., when the program card images are first put into a file on which SAIL can operate. Assume also that it is given the value 100. When SAIL is used to create a file for compilation, it will replace the expression:

```
      _XSIZE_
or    <XSIZE>
or    (XSIZE)
```

with the number 100 as long as the card on which the expression appears has a $ in column one. The $ is a flag which tells SAIL to scan the card for options delimited with _, < >, or ( ), signs. So the card:

```
$     COMMON / 1 / X(_XSIZE_)
```

would become:

```
      COMMON / 1 / X(100)
```

---

1. AFWL DYT TN 75-3, "THE SAIL UPDATE AND EXECUTIVE PROGRAM", Lewis P. Gaby, Daniel A. Matuska

2. AFWL TR 78-80, "SAIL, AN AUTOMATED APPROACH TO SOFTWARE DEVELOPMENT AND MANAGEMENT", Lewis P. Gaby, David C. Graham, Clifford E. Rhoades, Jr., Jan. 1979.

after SAIL processing. If the default value of 100 is not desired it can be changed in the SAIL processing input deck by using the commands:

SAIL OPTIONS XSIZE=50 ENDOPTIONS PROGRAM MAIN

This input deck tells SAIL to process program MAIN from the SAIL file and to change the value of the option XSIZE to 50. The option value can also be changed by another program which generates a file named INPUT2. This file contains the following cards:

```
PROGRAM MAIN
OPTION XSIZE 50
```

Thus XSIZE can be changed by a preprocessing program which, for example, reads the input deck for MAIN and decides how big the X array should be.

Options can be logical or numerical. They can be set as discussed before or by

*DEF cards in the file SAIL will update.

*DEFN XSIZE=50 will replace _XSIZE_ with a 50.

*DEFL XLOG=TRUE will change the value of the option named, for example, XLOG to TRUE.

The options provide powerful programming tools when used with *KEEPTO and *SKIPTO commands in the file for SAIL updating. The command:

*KEEPTO *36 XSIZE EQ 50

will tell SAIL to keep the next 36 cards for the compile file if XSIZE is equal to 50. The command:

*KEEPTO XEND XSIZE GT 50

will tell SAIL to keep all of the cards in the deck until it reaches a card:

*LABEL XEND

if XSIZE has a value greater than 50. Just as in FORTRAN commands, SAIL recognizes the operators EQ, GT, GE, LT, LE, and NE. Logical options can be used in a similar fashion; for example:

*KEEPTO *36 XLOG

will tell SAIL to keep the next 36 cards in the compile file if XLOG is TRUE.

*SKIPTO commands are used in a similar fashion but tell SAIL to skip coding - instead of keeping it. For example:

> *SKIPTO *36 XSIZE NE 50 tells SAIL to skip past the next 36 cards if XSIZE is not equal to 50.

Blocks of coding can be written once and then placed into a program at any number of points. If the block of coding is identified by any name with eight or less characters, say BLOCK1, then any place:

```
*INCLUDE BLOCK1
```

appears, SAIL will replace the card with the desired coding. A block of coding is given a name by the card:

```
*PROC NAME
```

where NAME could, for example, be BLOCK1. The coding must begin with the *PROC card and terminate with the card:

```
*ENDPROC
```

These procedures, or PROC's, are particularly useful in generating common blocks for insertion into several subroutines. For example, the following coding could be used to generate a common block.

```
*PROC        /TIME/
$       COMMON/TIME/X(_XSIZE_), Y(_YSIZE_)
*ENDPROC
```

The common block with option values placed in it will appear anywhere a:

```
*INCLUDE/TIME/
```

card appears. The // marks are not required, but can be used and are helpful in identification of a COMMON PROC.

A SAIL file can have many programs in it. Each program must begin with a:

```
*B NAME
```

card where NAME is the name of the program. If NAME is specified in SAIL input then only program NAME will be processed for compilation. There is one exception to this. The program named PROLOGUE will be processed each time any other program is processed. Thus coding common to many programs can be placed in PROLOGUE so that it does not have to be repeated in each program.

There are several programs in the EPIC SYSTEM file. They are as follows:

PROLOGUE
PREPSIZE
MAINSIZE
PREP
MAIN
POST1
POST2
EMON
HULLINSZ
HULLIN

PROLOGUE contains common blocks and routines used by several programs.

PREPSIZE is a program used to read PREP input and set option values in an INPUT2 file which SAIL uses when processing program PREP. That is, PREPSIZE determines the size of certain PREP arrays and which subroutines must be kept for the desired problem. PREPSIZE keeps the size of program PREP to a minimum. It also allows interactive debugging of decks since it executes the same commands as PREP with the exception that node and element files are not actually created.

MAINSIZE reads the input deck for program MAIN, the restart file made by program PREP or a previous MAIN run and a HULL input tape (if a linked HULL-EPIC run). From these data, it determines the size of arrays and sets options for deleting and maintaining coding in program MAIN.

P1SIZE reads the restart tape and time history tape (if used) and determines the size of arrays to be used in the plotting program POST1.

P2SIZE reads the time history tape and sets up options to size the plotting program POST2.

PREP is the EPIC generator program used when starting an EPIC problem. It sets up nodes and elements on a restart tape used by MAIN.

MAIN is the EPIC numerical physics program which solves the problem described on the restart tape.

POST1 is the program to make contour plots, vector velocity plots, etc., from data on the restart tape.

POST2 plots the time history data collected by MAIN on the time history tape.

EMON is a program which can be run interactively and allows two-way communication with a running EPIC program. Communication is provided via a permanent file.

HULLIN is a program to convert an EPIC restart tape to a format useable by HULL in overlaying an EPIC problem back into a HULL grid.

HULLINSZ is a sizing program for HULLIN.

PREP, MAIN, POST1, and POST2 have previously been fully described[3,4] except for changes made by OTI.

The options, their meaning and default values are described below:

OPTION	VALUES	MEANING	DEFAULT VALUE
	-- COMPUTER RELATED --		
COMPUTER	-1	EGLIN CYBER 176	1
	0	EGLIN CDC6600	
	1	BRL CDC7600	
	-- TAPE RELATED --		
	All tape options have the same meaning		
	0	means no file	
	1	means disk	
	7	means 7 track tape length	
	9	means 9 track tape length	
ITAPOT		Restart file always exists	9
ITAPLT	NON-ZERO	Time History tape in use	0
	0	Time History tape not in use	
TAPELIB	0	No automatic tape Library	0
	1	Automatic Tape Library in use (not yet implemented)	
CATPOT	1	Catalog ITAPOT file when full	1
	0	do not catalog	
CATPLT	1	Catalog ITAPLT file when full	1
	0	do not Catalog	

---

3. AFATL TR 78-81, "Further Development of the EPIC-3 Computer Program for Three-Dimensional Analysis of Intense Impulsive Loading", G. Johnson, Honeywell, Inc., July 1978.

4. "Further Development of EPIC-3 for Anisotropy, Siding Surfaces, Plotting and Material Models", G. Johnson, Oct. '79 (to be published as a BRL report)

Cataloging for ITAPOT and ITAPLT will occur when the file has reached the length defined by ITAPOT and ITAPLT or at the end of the current run -- whichever occurs first. If the file is cataloged because of its assigned tape length, a new file will be started and the run will continue. The files are cataloged as:

EPIC3TAPE9XXXXPTYYYY
and
EPIC3TAPE7XXXXPTYYYY

where the case number (now a floating point variable) for the problem being run is XXXX. YYYY. The files are cataloged in a directory determined by the USER option. At BRL, they are cataloged on the z machine.

On a restart the latest tape cycle is automatically attached and used to begin the problem. It is returned after the problem begins. So every restart will generate at least one more cycle of either tape. A given cycle of the restart file may contain more than one EPIC restart dump. It is up to the user to insure that he does not violate any local prohibitions on the total number of cycles cataloged.

CATPLOT	1	Catalog Tape99 from POST1 or POST2	1
	0	do not catalog	

The files are cataloged as:

P1XXXXPTYYYY
and
P2XXXXPTYYYY

where the P1 file is from POST1 and the P2 file is from POST2. They are cataloged under a directory determined by USER. At BRL the files are cataloged on the A machine.

USER		User works to determine directory Name for cataloging at BRL and Eglin AFB	

--- REMOTE CONTROL RELATED ---

RCONTRL	0	No remote control	0
	1	Remote Control is On	

The remote control option is explained in detail under the program EMON description:

--- VECTORIZED CODING RELATED ---

VECTOR	0	Vectorized coding not desired (only available for linked HULL runs)	0
	1	Vectorized coding desired	

--- DEBUG RELATED ---

DEBUG	0	No debug print desired	0
	1	Debug print desired in whatever program is being run	

--- PROBLEM RELATED ---

SLIDE	0	No sliding surface in the problem	1
	1	There is a sliding surface	
NHULL	0	No HULL input tape	0
	1	HULL tape required (Tape 14) for linked run	
EPICEOS	0	Normal EPIC metal equation of state (MTYPE=1 or 2) not used	1
	1	Normal EPIC metal equation of state is being used	
HULLEOS	0	The HULL equation of state (MTYPE=6) is not being used	0
	1	The HULL equation of state is being used	
HE	0	The EPIC high explosive equation of state (MTYPE=3) is not being used	0
	1	It is being used	
CONCRT	0	The EPIC concrete/geology equation of state (MTYPE=4) is not being used	0
	1	It is being used	
ISOTRP	1	The EPIC anisotropic equation of state (mtype=5) is not being used	1
	0	It is being used	
MATLIB	1	Obtain material properties from the material library. (not yet implemented)	0
	0	Input all material properties	
PYFAIL	0	Do not use a P/Y vs Strain to failure surface for failure determination	1
	1	Use a P/Y surface (PYFAIL can only be used with the HULL equation of state. If HULLEOS is 1, PYFAIL is automatically set to 1. If other materials are used in addition to those under the HULL equation of state, they will use the standard EPIC failure model)	

STRAIN	0	Do not save strains for elements	0
	1	Save total strains for elements (Total strains are required for the PYFAIL option. If PYFAIL=1, STRAIN components are automatically saved)	
FAIL	0	If failure occurs (ICHECK=1) do not change the element physics	0
	1	If failure occurs, do not let the element have strength or a tensile pressure capability. (FAIL only has meaning if the HULL equation of state is being used. The other EPIC equations of state will automatically change element physics if failure occurs)	
NROD	0	Subroutine NROD will not be used	1
	1	Subroutine NROD will be used	
NNOSE	0	Subroutine NNOSE will not be used	1
	1	Subroutine NNOSE will be used	
NPLATE	0	Subroutine NPLATE will not be used	1
	1	Subroutine NPLATE will be used	
NSPHER	0	Subroutine NSPHER will not be used	1
	1	Subroutine NSPHER will be used	
LROD	0	Subroutine LROD will not be used	1
	1	Subroutine LROD will be used	
LNOSE	0	Subroutine LNOSE will not be used	1
	1	Subroutine LNOSE will be used	
LPLATE	0	Subroutine LPLATE will not be used	1
	1	Subroutine LPLATE will be used	
LSPHER	0	Subroutine LSPHER will not be used	1
	1	Subroutine LSPHER will be used	
ILPLOT	0	Do not save internal loads data on tape ITAPLT	0
	1	Save internal loads data	
ISPLOT	0	Do not save system data on tape ITAPLT	0
	1	Save system data	
NNPLOT		The number of nodes at which time history data is desired (no longer limited to 20)	0

NLPLOT		The number of elements at which time history data is desired (no longer limited to 20)	0
ITIME	0	Time history data is not being saved	0
	1	Time history data is being saved	
RIGID	0	There is no rigid body in the problem	0
	1	There is a rigid body in the problem	
COMBINE	0	Do not combine projectile planes	0
	1	Combine planes (not yet implemented)	
LINK	1	The calculation is (or was) a linked run	0
	0	The calculation is not a linked run	

--- ARRAY SIZING AND STORAGE OPTIONS ---

NM		Number of materials in the problem	10
NPRES		Number of applied pressure (P,T) pairs	50
NVEL		Number of applied velocity (V,T) pairs	50
IPOY		Number of (P/Y, strain) pairs	10
LBSIZE		Number of elements in one band	64
LBAND		Number of element bands in temporary storage at one time	1
LASIZE		LBSIZE * LBAND = Number of elements in storage area at one time	
LBSP		LBSIZE * 6 + 3	
LBSPP		LBSIZE * 18 (LBSIZE * 24 if STRAIN=1)	
NEBLK		Number of element blocks in the problem (each block is LBSIZE long)	500
LCMEBUF	1	Element buffering arrays will be in LCM	0
	0	Element buffering arrays will be in SCM	
LCMELEM	1	/ELEMNT/and/VECTOR/ arrays and equivalenced arrays will be in LCM	0

	0	/ELEMNT/ and /VECTOR/ arrays and equivalence arrays will be in SCM	
NBSIZE		Number of nodes in one block	64
NBAND		Number of blocks in core	100
NBANDMN		Number of blocks required in core at one time (minimum value of NBAND)	0
NASIZE		NBSIZE & NBAND = number of nodes in temporary storage area at one time	
NBSP		NBSIZE + 2	
NBSPP		NBSIZE * 14	
NNBLK		Number of blocks in the problem	
LCMNBUF	1	Node buffering arrays will be in LCM	0
	0	Node buffering arrays will be in SCM	
LCMNODE	0	The /NODE/ common blocks will be in SCM (if NBAND is set to NNBLK all nodes will be in SCM)	0
	1	The /NODE/ common blocks will be in LCM and NBAND is set to the minimum required to run the problem	
	2	The /NODE/ common blocks will be in LCM and NASIZE is automatically set to NNBLK * NBSIZE; that is, all nodes will be in LCM (NBAND=NNBLK)	
NSTN		The number of time history stations on the HULL input tape (if NHULL=1)	

--- OUTPUT OPTIONS ---

IPRPLOT	1	A printer plot of the geometry will be produced whenever a dump is made (in PREP and MAIN)	1
	0	A printer plot will not be produced	

The number of SAIL options is rather large. However, most are set automatically by the sizing program based on data in input decks and the values of other options. The few options actually specified by the user normally need be set only once--at the time PREP is run. All option values are automatically saved in records on both the restart and time history tapes. Therfore, unless an option value is being changed it normally does not need to be specified again by the user. If an option

is not specified by the user or set by a sizing program it will automatically assume its default value. And the defaults have been set to reasonable values, so a user can simply run without setting any options and be guaranteed a "good" run insofar as a "good" run is controlled by option values.

USER, CATPOT, CATPLT and DEBUG are exceptions to the rule that options need not be reset. The sizing program must be able to find Tape 7 and Tape 9 to find the option table. Since default values of CATPOT and CATPLT are set to 1 - i.e., files are assumed to be cataloged - the sizing program must be given the correct USER number if the files are indeed catalogued. If the files are not cataloged then CATPOT and CATPLT must be set to zero in each sizing program SAIL deck, so that the sizing program does not attempt to attach non-existent files. DEBUG is reset to zero by MAINSIZE, P1SIZE and P2SIZE prior to reading new option values.

We will describe how each option is set during a sequence of PREP, MAIN, POST1, and POST2 runs and we will indicate which options can be changed by the user. Prior to this, however, it will be helpful to understand the control deck structure for running EPIC System programs. The basic structure is seen below for the Z machine at BRL.

Job Card	
Account Card	
ATTACH, HULLIB, ID=KIMSEY. LIBRARY, HULLIB.	Brings in HULLIB Library
ATTACH, CHEPIC, ID=KIMSEY.	Brings in LATEST change deck for EPIC
COPYS, INPUT, TAPE5.	Puts input deck on Tape5
REWIND, TAPE5.	
COPYS, INPUT, A. COPYS, CHEPIC, A. REWIND, CHEPIC, A.	Puts sizing program SAIL record and EPIC changes on local file A
COPYS, A, TAPE10. REWIND, A, TAPE10.	Puts sizing program SAIL record on local file TAPE10

FILE, OLD, RT=S, BT=C. GETPF, OLD, EPICSYS, ID=KIMSEY, ST=MFA.	Brings in EPIC System file
DYTHUL, I=A.	Executable SAIL is DYTHUL. This command updates EPIC and makes a file called SAIL ready for compilation
FTN, I=SAIL, B=SIZE.	Compiles file SAIL to execute as program SIZE

ATTACH or REQUEST the restart (Tape 9) and time history (Tape 7) tapes if not running under automatic cataloging.

SIZE.	Executes program SIZE
RETURN, SAIL, TAPE10, SIZE, A.	
COPYS, INPUT, B. COPYS, CHEPIC, B.	Brings in primary program SAIL update record and combines it with EPIC change deck
REWIND, B. RETURN, CHEPIC. FILE, ØLD, RT=S, BT=C. GETPF, OLD, EPICSYS, ID=KIMSEY, ST=MFA.	Old was returned when SAIL was executed. It must be brought in again
DYTHUL, I=B.	SAIL is executed to update the EPIC System file
FTN, I=SAIL, B=MAIN.	Compiles file SAIL
RETURN, SAIL, B. MAIN.	Executes primary program

Add any desired cataloging cards here if not using automatic cataloging

```
*EOR
   input data deck
```

```
*EOR
   SAIL OPTIONS USER=10
     ENDOPTIONS

   PROGRAM SIZE
```

SAIL UPDATE DECK. Word SIZE should be PREPSIZE, MAINSIZE, P1SIZE, P2SIZE or HULLINSZ as appropriate

```
*EOR
   SAIL PROGRAM MAIN
```

SAIL update deck for the primary program. The word MAIN should be PREP, MAIN, POST1, POST2, or HULLIN as appropriate.

As seen in the deck structure, the only location for changing option values is in the SAIL update record for the sizing program. Any option changes in the primary program update record are ignored. Each sizing program calls a subroutine CHOPT which reads TAPE10 and changes option values through the INPUT2 file it creates. It should be noted that this is somewhat different than the situation which exists in running the HULL program. In EPIC, all but derived options (such as LBSP, NBSP, etc.) are saved in an /OPTION/ common block which is put on Tape9 and Tape 7. The sizing program reads this common block and puts it on INPUT2 after changing any values according to the sizing program SAIL update record. When the primary program is being built by SAIL, any option values on INPUT2 will override option values set in the SAIL update record for the primary program. Since essentially all option values are specified in INPUT2, were it not for CHOPT any options set in the sizing or primary program SAIL update records would be, in effect, ignored. The HULL program does not specify all of its options in INPUT2 and many options can be set in the primary program SAIL deck. There are advantages to both systems. The advantage in the system chosen for EPIC is deck simplicity. Once options have been set in the PREP run, they need never be set again with the exception of USER and possibly CATPOT and CATPLT; whereas in HULL some options have to keep being reset or they will revert to default values.

We are now in a position to discuss how options get set, automatically and otherwise, during a full EPIC run sequence.

The user first runs PREPSIZE and PREP. PREP will expect to have USER, LBSIZE and NBSIZE specified. (It actually expects to have LBAND also specified but the default option of 1 is a good value for all runs). LBSIZE and NBSIZE will assume values of 64 if not given other values in the PREPSIZE SAIL record. PREPSIZE reads the PREP input deck and automatically sets:

CONCRT, EPICEOS, HE, HULLEOS, ISOTRP, IPOY, LNOSE, LPLATE, LROD, LSPHER, NNOSE, NPLATE, NROD, NSPHER, NM, NASIZE, NBANDMN, NBAND, NEBLK, NNBLK, NPRES, NVEL, NHULL, LINK, NSTN, PYFAIL, STRAIN, SLIDE, and RIGID.

(NBANDMN is the minimum required to run the problem)

Default values will be assumed for:

COMPUTER, ITAPOT, ITAPLT, CATPOT, CATPLT, DEBUG, FAIL, LCMEBUF, LCMELEM, LCUMNBUF, LCMNODE, RCONTRL, VECTOR, USER and IPRPLOT

unless otherwise specified by the user at this time. Values specified by the user at this time will remain with the problem for all subsequent runs with the exception that LCM availability will be checked by each sizing program and LCM options will be reset if there is not enough available LCM storage. At BRL, we assume 363000_{10} words of LCM availability. Options set from the data input deck should not be changed by the user, with the exceptions of NHULL, COMBINE and NBAND or NASIZE. PREPSIZE sets NBAND and NASIZE (=NBAND * NBSIZE) to the smallest values possible for running -- i.e., it minimizes core storage. If the user desires to run with all node storage in LCM, then he must set LCMNODE to 2 on the PREP or subsequent MAIN runs.

Program MAINSIZE reads the SAIL options from the first dump on the restart tape and examines MAIN input data to set:

ISPLOT, ILPLOT, NNPLOT, NLPLOT, ITIME, NPRES (if IPRES=2), NHULL and NSTN.

It also checks desired LCM options against LCM availability and resets them if required.

Program P1SIZE reads the first dump on the restart tape to optain SAIL options for the run. It automatically sets ITIME=1 and resets LCM options if required.

Program P2ISZE reads the time history tape to optain SAIL options for the run. It also sets ITIME=1.

SECTION IV
THE HULL EQUATION OF STATE AND STRESS/STRAIN RELATIONS

A new metal equation of state (MTYPE=6) was added to EPIC. It is termed "the HULL equation of state" simply because it is modeled after that used in the HULL code. The use of this equation of state will insure the consistency required for linked HULL/EPIC runs. In addition, the equation of state and associated stress/strain relations are considered more physically appropriate than the other EPIC metal relations. Since HULL only runs in CGS units, an EPIC linked run must also employ CGS units. In an unlinked EPIC run, this HULL equation of state may be used with any consistent set of units.

Input for a HULL Metal is as follows:

- ρ_o - initial density
- C_o - initial bulk sound speed
- s - shock velocity/particle velocity slope
- Γ_o - initial Grueisen ratio
- PMIN - maximum tensile pressure (normally set to a very large negative number)
- ν - Poisson's ratio
- Q1 - artificial viscosity coefficient
- Q2 - artificial viscosity coefficient
- FY - initial yield strength
- FU - ultimate yield strength
- EU - strain at ultimate yield
- EMELTO - energy density at onset of melt at ambient density
- EFUS - fusion energy density
- YF1, EF1, YF2, EF2 - Thermal softening curve parameters
- NPOY - number of (P/Y, E_F) pairs
- (POY,EPOY) - (P/Y, E_F) pairs
- EBS - maximum tensile strain for dropping an element from the grid (i.e., setting ICHECK to -1)

Each of these input variables and the equation of state logic will be explained below:

PRESSURE, ENERGY AND ARTIFICIAL VISCOSITY CALCULATIONS

The pressure, sound speed, internal energy and artificial viscosity are calculated in subroutine PHULL. The calculation proceeds as follows:

The Hugoniot pressure, P_H, is calculated from:

$$P_H = C\mu + D\mu^2 + S\mu^3 \qquad \text{if } \mu \geq 0$$
$$= C\mu \qquad \text{if } \mu < 0$$

where the excess compression, μ, is defined by:

$$\mu = \rho/_{\rho o} - 1$$

and C, D and S are calculated from:

$$C = \rho o\ Co^2$$
$$D = C\ (1+2(s-1))$$
$$S = C\ (2(s-1)+3((s-1)^2)$$

The current Gruneisen ratio is computed from:

$$\Gamma = \Gamma_o\ \rho_o/\rho$$

with a maximum value set at $2\Gamma_o$ for very highly distended elements. This representation of Γ is generally considered more appropriate than $\Gamma = \Gamma_o$ for metals. In addition, if Γ is a constant for all densities, maxima will exist in the total pressure computation for relatively small values of μ. P_H and Γ are then used to form:

$$F1 = P_H\ (1 - \Gamma\ \mu/2)$$

and

$$F2 = \Gamma\rho = \Gamma o \rho o$$

which eventually will be combined to form the Mie-Gruneisen total pressure:

$$P = F1 + F2 \cdot E$$

when E is the element's internal energy density.

Prior to combining F1 and F2 to form the element's pressure, these quantities are subjected to two checks to keep them at reasonable levels for very highly expanded elements.

If $-0.25 < \mu < -0.2$, then F1 is varied from its value at $\mu = -0.2$ to zero at $\mu = -0.25$. The theory behind this check is simply that any metal expanded to a density of 0.8 times initial density must have failed and an element with such conditions must consist of highly fractured metal particles with no overall tensile capability.

If $-0.4 < \mu < -0.25$, then F2 is varied from its calculated value at $\mu = -0.25$ to zero at $\mu = -0.4$. The purpose of this check is to prevent extremely expanded elements from existing at high pressure levels due to large internal energy values.

These two checks taken together keep an element from maintaining high tensile or compressive pressures when it is in a very higly expanded state. The F1 check provides the type of reasonable tensile pressure cutoff normally provided by a PMIN check. Thus PMIN is redundant and is only included in the equation of state to provide flexibility for the user.

After the F1 and F2 checks are complete, pressure is computed from:

$$P = F1 + F2 \cdot E$$

and a sound speed squared is computed from:

$$SS2 = \frac{\partial (F1)}{\partial \rho} + P \cdot (F2)/\rho^2 + (4/3)G/\rho o$$

where G is the shear modulus.

The equation for SS2 takes advantage of the thermodynamic identity

$$\left(\frac{\partial P}{\partial \rho}\right)_s = C^2 = \left(\frac{\partial P}{\partial \rho}\right)_E + \frac{P}{\rho^2} \left(\frac{\partial P}{\partial E}\right)_\rho$$

where subscript s refers to entropy. SS2 is then used to compute the artificial viscosity from the standard EPIC relationship.

After these computations, P is checked against PMIN. If P is less than PMIN, both P and E are reset. If the SAIL option FAIL is set to 1, the equation of state checks P and limits it to compressive values if ICHECK = 1. As a last check on P, the equation of state compares the computed P to a variable PMAX, now set at 100 megabars. Values of pressure this large should not be exceeded if the code is running stably.

STRESS/STRAIN CALCULATIONS

Subroutines involving stress and strain calculations were charged to make input simpler, to incorporate a physically realistic failure model and a thermal softening model consistent with that used in the HULL code.

The shear modulus, G, is now computed from:

$$G = \frac{3}{2}\frac{(1-2\nu)}{(1+\nu)}K_0$$

where ν is the input value of Poisson's ratio and K_0 is the initial bulk modulus, $\rho o C o^2$.

The non-thermally softened yield strength is calculated from:

$$Y = F_Y + (F_U - F_Y) \cdot \varepsilon_P / \varepsilon_U \text{ if } \varepsilon_P \leq \varepsilon_U$$

$$Y = F_U \text{ if } \varepsilon_P > \varepsilon_U$$

where ε_P is the accumulated plastic strain for the element.

This value of Y is then multiplied by a thermal softening factor which varies from 1 when the internal energy density is 0 to 0 when the internal energy density is:

$$E_M + 0.5 \cdot E_{FUS}$$

Where E_M is the onset of melt energy at current density given by:

$$E_M = E_{M0} \cdot (1 + C6 \cdot \eta + C7 \cdot \eta^2) \quad (5)$$

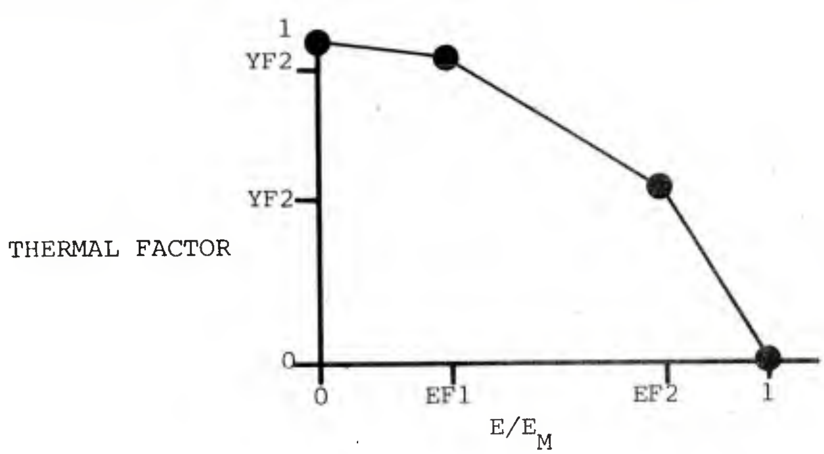
where E_{M0} is the energy density at the onset of melt at ambient density.

C6 and C7 are material coefficients calculated from:

$$C6 = 2\ \Gamma o - 2/3$$
$$C7 = (\Gamma o - 1/3)\ (2 \cdot \Gamma o + 1/3) - 1$$
$$\eta = \text{volumetric strain} = 1 - \rho o/\rho$$

and E_{FUS} is the energy density required for completion of melt at ambient density.

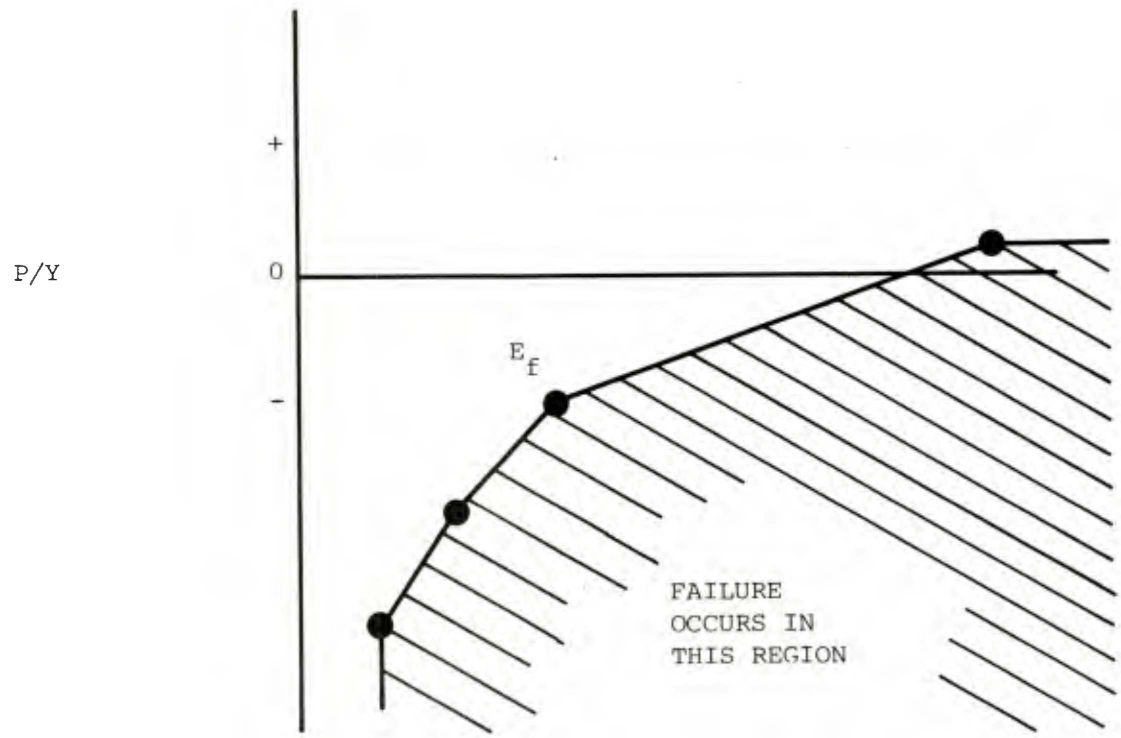
The thermal factor is described by means of a linear fit between four points as shown below:



The points (EF1, YF1) and EF2, YF2) are input values. If only one point is desired, both points should be input as the same values.

There is no explicit strain rate factor used in the yield strength calculation. It is assumed that the material will be deformed at strain rates on the order of 100 sec^{-1} or greater. The yield strength of most metals does not change appreciably with strain rate beyond this point.

The failure model used with a HULL metal in EPIC is a generalized model which should correctly predict failure for any state of stress. The user defines a table of P/Y vs plastic strain at failure points. The type of failure surface generated should generally appear as seen below (if all states of stress are to be considered):



5. Richard Grover, "Liquid Metal Equation of State Based On Scaling", Journal of Chemical Physics, Vol. 55, No. 7, 1 October 1971

The curve illustrated above is developed from four input pairs. The basic theory behind such a curve is that failure will occur at small values of strain for essentially plane strain states (tensile P/Y values of -1 to -2), at medium values of strain for plane stress states (tensile P/Y values from -0.2 to -0.4) and at very large values of strain for compressive stress states. The strain used for the model is the maximum principle plastic strain if this strain is positive - i.e., if it is tensile.

The model requires calculation of the maximum principle plastic strain, which means that current strains must be known for each element. If the HULL equation of state is used, the SAIL option STRAIN is automatically set to 1 and the isotropic strains ε_{XX}, ε_{YY}, ε_{ZZ}, ε_{XY}, ε_{XZ}, and ε_{YZ} are saved for each element. The maximum principle strain is computed by finding the maximum root, E, of the determinant equation:

$$\det \begin{pmatrix} \varepsilon_{XX} - \varepsilon & \varepsilon_{XY} & \varepsilon_{XZ} \\ \varepsilon_{XY} & \varepsilon_{YY} - \varepsilon & \varepsilon_{YZ} \\ \varepsilon_{YZ} & \varepsilon_{XZ} & \varepsilon_{ZZ} - \varepsilon \end{pmatrix} = 0$$

this determinant equation reduces to:

$$\varepsilon^3 + A \cdot \varepsilon^2 + B \cdot \varepsilon + C = 0$$

where:

$$\begin{aligned}
A &= -\varepsilon_{XX} - \varepsilon_{YY} - \varepsilon_{ZZ} \\
B &= \varepsilon_{XX} \ \varepsilon_{ZZ} + \varepsilon_{XX} \ \varepsilon_{YY} + \varepsilon_{YY} \ \varepsilon_{ZZ} \\
&\quad -\varepsilon^2_{XY} - \varepsilon^2_{XZ} - \varepsilon^2_{YZ} \\
C &= -\varepsilon_{XX} \ \varepsilon_{YY} \ \varepsilon_{ZZ} + \varepsilon_{XX} \ \varepsilon^2_{YZ} \\
&\quad + \varepsilon_{YY} \ \varepsilon^2_{XZ} + \varepsilon_{ZZ} \ \varepsilon^2_{XY} \\
&\quad + \varepsilon_{YY} \ \varepsilon_{XZ} + \varepsilon_{ZZ} \ \varepsilon^2_{XY} \\
&\quad - 2 \ \varepsilon_{XZ} \ \varepsilon_{YZ} \ \varepsilon_{XY}
\end{aligned}$$

Substitution of $\varepsilon' = \varepsilon + A / 3$ reduces the equation to:

$$(\varepsilon')^3 + a \ \varepsilon' + b = 0$$

where:

$$a = (3B - A^2) /3$$

and

$$b = (2A^3 - 9AB + 27C) /27$$

If a = 0 and b = 0, then all three roots are equal and are

$$\varepsilon' = 0$$

If a = 0 but b $\neq$ 0, then all three roots of the equation are the same and are given by:

$$\varepsilon' = b^{1/3}$$

If a $\neq$ 0 the three roots are:

$$\varepsilon_1' = F \cos (\phi/3)$$

$$\varepsilon_2' = F \cos (\phi/3 + 2\pi/3)$$

$$\varepsilon_3' = F \cos (\phi/3 + 4\pi/3)$$

where $F=2\sqrt{-a/3}$ and $\cos \phi = -(b/2) / \sqrt{-a^3/27}$. Of these three roots the first will always be the largest or equal to the largest.

Once the largest root of ε' is found, the maximum principle strain is found by inverting the previous substitution:

$$\varepsilon = \varepsilon' - A/3$$

The maximum principle plastic (deviatoric) strain is the principle strain reduced by -A/3. Note that this quantity is ε' before it is converted. We employ maximum plastic strain simply because it is the quantity measured in tests. Spherical elastic strain can contribute to failure, but it is not present in the sample when the tests are complete and the measurements taken.

If IPNODE is set to 1 in PREP then both P/Y and maximum tensile strain used for the failure check are averages of the values at the element's nodes and these nodal values are themselves averages of the values in all of the elements surrounding each node. Use of this option smooths EPIC data with the possible reduction of peak values.

It should be noted that EPIC does not save internal energy density (energy divided by mass) for an element. It saves energy divided by original element volume. Thus in implementing the equations presented in this section, division of stored energy by ρo was required in several locations to convert it to energy density.

SECTION V
EPIC GENERATOR CODING CHANGES AND INPUT INSTRUCTIONS

In this section we will discuss specific coding changes made to PREP to implement the link and other improvements. We will also discuss the new PREP sizing programs. Since the coding changes do not affect the structure of PREP new flow charts will not be presented.

The PREPSIZE program is run first. PREPSIZE is really almost a version of PREP but without any large arrays. Node and element geometries are generated to the extent required to determine parameters such as the number of nodes and elements in the problem. The primary purpose of this PREP simulation is the collection of data needed to size PREP arrays, establishment of values of NBAND, NNBLK, etc. Secondarily, the exercise provides significant debug of a PREP input deck since it goes through almost all of the PREP instructions. Since it does not use any large arrays it can be run interactively if desired as a PREP deck debug program. Normally, of course, it is run in a command chain which includes the PREP program.

Input to PREPSIZE (and all of the programs in the EPIC system) is now format free. Thus input decks can be created with a minimum of effort on a remote terminal and batched to the main computer. The routine used is subroutine READIN, contained in PROLOGUE. READIN uses the HULL library routine VALUE to crack the input file into separate input words or values. Each input word is read with a separate call to READIN. READIN always assumes the input variable is floating point (even though there may not be a decimal point in it) so a separate statement is required to convert it to integer. Words in an input deck are separated by spaces or commas. Input can be in any of the forms below:

```
X
X.Y
X.EY
X.E + Y
X.E - Y
```

If there is no decimal - as in the first example - READIN assumes one at the end of the word. The number of characters represented by X or Y is limited only by the computer's capabilities. READIN cannot be used for alphanumeric data. Separate READ statements are employed for such input.

PREPSIZE first prints all input deck cards with a call to the PROLOGUE routine INOUT. Then the input deck is rewound and processing begins. It reads and processes all of the cards to the point that it can establish values for the SAIL options:

NEBLK
NBAND
NBANDMN
NNBLK
SLIDE
NM
IPOY
HULLEOS
EPICEOS
HE
CONCRT
ISOTRP
PYFAIL
NHULL
LINK
NVEL
RIGID

The sail options NBSIZE, LBSIZE and LBAND are required to establish NBAND, NNBLK, etc. Default values are used if new values are not specified in the PREPSIZE SAIL deck. From the input deck, PREPSIZE also determines values for SAIL options:

LROD
LNOSE
LSPHER
LPLATE
NROD
NNOSE
NSPHER
NPLATE

which are used as flags in PREP construction to determine if the element and node generation routines of the same names are required. Other options - such as USER, LCMNODE, etc. are assumed to retain default values unless they are given new values in the PREPSIZE SAIL deck.

All of the option values established by PREPSIZE are written into the INPUT2 SAIL file so that they are available when SAIL creates the PREP source file.

If DEBUG is set to 1 in the PREPSIZE SAIL deck, PREPSIZE will print all of the data normally printed by PREP as it processes input. In addition, it prints node geometry data for any nodes with non-zero values of IXYZ. IXYZ is a node word which contains spatial restraint, sliding surface and a node velocity driving flag. An extra digit (in the 10 thousands place) was added to IXYZ as part of the link implementation. This digit, with the variable name IVEL, is 1 if the node is driven by HULL input, 2 if the node is driven by input velocity cards and 0 if the node is not externally driven.

At the time PREPSIZE is run, the user should at a minimum, give some thought to values desired for the SAIL options:

USER
ITAPOT
CATPOT
ITAPLT
CATPLT
FAIL
STRAIN
LCMELEM
LCMEBUF
LCMNODE
LCMNBUF

These options were discussed in detail in a previous section of this report. The default values of these variables will provide an adequate run but they may not provide the most efficient or physically accurate run. The LCM options can be changed at any time if desired. The other options listed above should not be changed once they are set in the PREPSIZE run.

By reading and processing the PREP input deck, PREPSIZE is able to determine minimum values for NBAND, NASIZE and other variables previously estimated by the user or left at very large values which required PREP to use excess memory. Through SAIL and the PREPSIZE and PREP command chain, these variables are now automatically determined and implemented. PREP then will run at the smallest size possible for the problem being generated.

The variables required in PREP input are presented below. Detailed variable discussions are not presented for variables documented in previous EPIC reports.

PREP INPUT

DESCRIPTION CARD

PROBLEM DESCRIPTION, 8A10

MISCELLANEOUS CARD

CASE, ISAVE, IPRINT, ISPLIT, IPNODE*,N1R, NNR IZR, N1L, NOR, NIR, NPL

MATERIAL DATA

NUMBER AND TYPE

M, MTYPE

DESCRIPTION

MATERIAL DESCRIPTION (10A6)

IF MYTYPE = 1, EPIC METAL

DENSITY, SPH HEAT, k1, k2, k3, Γ, C_L, C_Q
SHEAR MOD, YIELD, ULT, $\bar{\varepsilon}$-ULT, PMIN, CR1, CT1, CT2
ε_v -SHEAR, $\bar{\varepsilon}$-SHEAR, $\bar{\varepsilon}$-TOTAL, TEMP

IF MTYPE = 2, EPIC LIQUID METAL

DENSITY, SPH HEAT, k1, k2, k3, Γ, C_L, C_Q
$\bar{\varepsilon}$-TOTAL, PMIN, TEMP

*NOTE: If IPNODE is set to 1 for a run involving MTYPE=6 material, nodal P/Y and strain to failure values are used in computing failure.

IF MTYPE = 3, HIGH EXPLOSIVE

DENSITY, ENERGY, DET VEL, C_L, C_Q, $\bar{\varepsilon}$-TOTAL
C1, C2, C3, C4, C5

IF MTYPE = 4, CONCRETE/GEOLOGICAL

DENSITY, SPH HEAT, TEMP, ε_v-SHEAR, $\bar{\varepsilon}$-SHEAR, $\bar{\varepsilon}$-TOTAL, C_L, C_Q
SHEAR MOD, YIELD, ULT, PMIN, CR1, CT1, CT2
C1, C2, C3, C4, C5, C6, C7, C8
C9, C10, C11, C12, C13, C14, C15, C16

IF MTYPE = 5, ANISOTROPIC MATERIAL

DENSITY, SPH HEAT, k1, k2, k3, Γ, C_L, C_Q
YIELD, ULT, ε-ULT, PMIN, TEMP, ε_v-SHEAR, $\bar{\varepsilon}$-SHEAR, $\bar{\varepsilon}$-TOTAL
Θ_1, Θ_2, Θ_3, E_x, E_y, E_z
G_{xy}, G_{yz}, G_{zx}, ν_{xy}, ν_{yz}, ν_{zx}
X-YIELD, X-ULT, ε_x-ULT, Y-YIELD, Y-ULT, ε_y-ULT
Z-YIELD, Z-ULT, ε_z-ULT, XY-YIELD, XY-ULT, γ_{xy}-ULT
YZ-YIELD, YZ-ULT, γ_{yz}-ULT, ZX-YIELD, ZX-ULT, γ_{zx}-ULT

IF MTYPE = 6, HULL METAL

DENSITY, SOUND SPEED, S, Γ, PMIN, ν, C_L, C_Q
YIELD, ULT, $\bar{\varepsilon}$-ULT, E_m, E_{fus}, YF1, EF1, YF2, EF2
NPOY, P/Y(1), $\varepsilon_f(1)$, ..., P/Y(NPOY), ε_f(NPOY)
ε_{MAX}

S = shock velocity/particle velocity slope

ν = Poisson's Ratio

E_m = melt energy density

E_{fus} = fusion energy density

YF1, EF1, YF2, EF2 = softening coefficients

ε_{MAX} = max tensile strain allowed before dropping element from grid

TERMINATION CARD

0

PROJECTILE SCALE/SHIFT/ROTATE CARD

XSCALE, YSCALE, ZSCALE, XSHIFT, ROTATE, SLANT

Node data cards for projectile (description follows)

TERMINATION CARD

0

TARGET SCALE/SHIFT/ROTATE CARD

XSCALE, YSCALE, ZSCALE, XSHIFT, ZSHIFT, ROTATE, SLANT

Node data cards for Target (description follows)

TERMINATION CARD

0

Element data cards for projectile (description follows)

TERMINATION CARD

0

Element data cards for Target (description follows)

TERMINATION CARD

0

SLIDING SURFACE CARDS:

MGEOM, ISEEK, IT, IM1, IS1, ISN, NSLV, NSG, ISR, FRICT, VREF, DREF

MASTER CARD FOR PLATE GEOMETRY (MGEOM=1)

NML, NMW, IDL, IDW, IDIA

MASTER CARD FOR NOSE-ROD GEOMETRY (MGEOM=2)

NOR, NIR, NPL

MASTER CARD FOR DISK GEOMETRY (MGEOM=3)

NRING, MCODE

MASTER CARD FOR CYLINDER GEOMETRY (MGEOM=4)

NRING, NPL, IDL, MCODE

MASTER CARD FOR ERODING DISK GEOMETRY (MGEOM=5)

NRING, NSURF, IDT, ELE1, $\bar{\varepsilon}$-max, $\bar{\varepsilon}$-ave, ΔT CHECK

INDIVIDUAL SLAVE NODE CARD(S) - FOR NSG=0

IS1, IS2, ..., ISN

NSG GROUPED SLAVE NODE CARD (S)

IS1G, ISNG, INC

TERMINATION CARD

0

DETONATION CARD

XDET, YDET, ZDET

INITIAL VELOCITY CARD

PXDOT, PYDOT, PZDOT, TXDOT, TYDOT, TZDOT, DT1

NODE INPUT DATA

LINE OF NODES CARD

1, NNODE, X1, Y1, Z1, XN, YN, ZN, EXPAND, IX, IY, IZ

ROD NODE CARDS

IDENTIFICATION

2

ROD NODE DESCRIPTION CARD

NOR, NIR, NPLN, IRAD, ZTOP, ZBOT, EXPAND, NHSURF, NHFRNT

ROD NODE TOP RADII CARD FOR IRAD=0

ROTOP, RITOP, ROBOT, RIBOT

ROD NODE TOP RADDI CARD(S) FOR IRAD=1

RT(NIR), ..., RT(NOR)

ROD NODE BOTTOM RADII CARD(S) FOR IRAD=1

RB(NIR), ..., RB(NOR)

ROD NODE TOP SURFACE CARD FOR IRAD=1

ZT(NIR), ..., ZT(NOR)

ROD NODE BOTTOM SURFACE CARD FOR IRAD=1

ZB(NIR), ..., ZB(NOR)

NHSURF	= 0 means nodes on the surface of the cylinder will not be driven by input velocity
	= 1 means that all cylindrical surface nodes will be driven by HULL input
	= 2 means that all cylindrical surface nodes will be driven by velocity input
NHFRNT	= 0 means that no front surface nodes will be driven by input velocity
	= 1 means that all front surface nodes (including those on the front face at the outer cylindrical surface) will be driven by HULL input
	= 2 means that all front surface nodes will be driven by an input velocity

NOSE NODE CARDS

IDENTIFICATION

3

DESCRIPTION

NOR, NIR, INOSE, IRAD, ROTOP, RITOP, ZTOP, ZMIN, NHSURF

NOSE NODE TOP RADII CARD(S) FOR IRAD=1

RT(NIR), ..., RT(NOR)

NOSE NODE ZMIN CARDS FOR IRAD=1

ZM(NIR), ..., ZM(NOR)

NOTE: if NIR=0, begin top radii and ZMIN cards with RT(1) and ZM(1)

<u>NHSURF</u> = 0 means that no surface nodes are driven by external velocities

= 1 means that all surface nodes are driven by HULL input

= 2 means that all surface nodes are driven by a velocity input

FLAT PLATE NODE CARDS

IDENTIFICATION

4, TYPE

DESCRIPTION IF TYPE=1

NX, NY, NZ, NXEND, NYEND, IY, X-EXPAND, X-PART, Y-EXPAND, Y-PART, Z-EXPAND

DESCRIPTION IF TYPE=2

NX, NY, NZ, NXEND, NZEND, IY, X-EXPAND, X-PART, Z-EXPAND, Z-PART, Y-EXPAND

NODE SIZE CARD

X1, Y1, Z1, XN, YN ZN

SPHERE NODE CARDS

IDENTIFICATION

5

DESCRIPTION

NOR, NIR, IRAD, RO, RI, ZCG

NODE RADII FOR IRAD=1

R(NIR), ..., R(NOR)

NOTE: if NIR=0, begin card with R(1)

ELEMENT INPUT DATA

COMPOSITE ELEMENT CARD

1, NCOMP, MATL, N1, N2, N3, N4, N5, N6, N7, N8, INC

ROD ELEMENT CARDS

IDENTIFICATION

2

DESCRIPTION

NOER, NIER, NLAY, N1, IMAT, MATL

MATERIAL IDENTIFICATION FOR IMAT=1

M(NIER), ..., M(NOER)

NOSE ELEMENT CARDS

IDENTIFICATION

3

DESCRIPTION

NOER, NIER, N1, IMAT, MATL

MATERIAL IDENTIFICATION FOR IMAT=1

M(NIER), ..., M(NOER)

FLAT PLATE ELEMENT CARDS

IDENTIFICATION

4, TYPE

DESCRIPTION

NLX, NLY, NLZ, N1, MATL

SPHERE ELEMENT CARDS

IDENTIFICATION

5

DESCRIPTION

NOER, NIER, N1, IMAT, MATL

MATERIAL IDENTIFICATION FOR IMAT=1

M(NIER), ..., M(NOER)

SECTION VI
EPIC MAIN CODING CHANGES AND INPUT INSTRUCTIONS

Program MAINSIZE is run before MAIN. MAINSIZE reads the input deck the header blocks from the restart tape and the HULL input tape (if there is one) and sets up SAIL options for processing MAIN. It also checks desired LCM options against availability and resets them if there is not enough LCM.

Program MAIN has been modified extensively. Incorporation under SAIL has been previously discussed as has the HULL metal equation of state. Of course, input is in field free format. The major changes to MAIN will be discussed in detail below.

A. SPECIFYING DUMP TIME

Input specifying the time at which restart dumps are to be made has been simplified. The method of providing a card for each dump can still be used. However, if only a single card is provided, the problem time specified on that card is treated as a time increment. Of course, problem termination always results in a dump unless ISAVE is set to zero.

B. PROBLEM TERMINATION

Problem termination can now occur in several ways. In addition to problem time exceeding TMAX or CP time exceeding CPMAX, or DT exceeding DTMIN a run can be terminated by turning on Sense Switch 1, through CP time about to exceed job card time limit, through the Monitor routine or when the minimum Z coordinate of the projectile reaches an input ZSTOP value (for linked run only).

C. RESTART FILE FORMAT

The restart file format has been changed slightly. New HULL link, equation of state and SAIL variables are now dumped. Users requiring details of each record on the tape can find this information in Subroutine SAVE in the PROLOGUE program.

D. CASE NUMBER

Case or problem number is now a floating point variable. This provides the user with added flexibility in problem identification. Case number is kept on the restart tape and any time history tapes. A comparison of requested case number is made with the case number on such files prior to allowing a restart or plotting to occur.

E. PROBLEM DESCRIPTION

The problem description array (DESC) is now kept on the restart and time history tapes. The user is therefore not required to re-enter it when restarting or plotting a problem.

F. TIME HISTORY DATA

MAIN has been rewritten to accept (X, Y, Z) positions for element and node time plots, to choose its own time increment for these plots and to pack data into fewer records eliminating many lengthy end-of-record areas.

If an element or node for time plotting is input as a zero, the code expects X, Y, Z values to follow. After all time history input has been read, all zero node and element inputs are filled in by a search through the grid. Node numbers are determined by finding the node at the minimum distance from the desired (X, Y, Z) point. Element numbers are determined by finding the element within which the desired (X, Y, Z) points reside. This is accomplished very simply by calculating the volumes of the four tetrahedra defined by the point and the element's four planes. If these volumes add up to the volume of the element then the point is within or on the surface of the element. After all node and element numbers have been found, they are ordered before being placed in the NPLOT and LPLOT arrays. These arrays are now sized by SAIL based on the requested input so there is no artifical limit imposed on how many time history points can be specified. The logic for determining time history node and element numbers is found in subroutine TOUT in PROLOGUE.

The user no longer inputs a time increment at which node and element time history data should be saved. Node data is saved any time that the node's velocity has changed by two percent from the last saved velocity. Element data is saved whenever the element's mean stress has changed by two percent. This percentage is set in data statements in subroutines LPLOTS and NPLOTS and can be adjusted to provide more or less data. This technique insures a minimum amount of data to define all details of a complete time history trace and is based on the time history technique employed in HULL.

Data is placed in records approximately 1,000 words in length. When the array ATIME (1024) would be over-filled by the next saved point, it is dumped to tape and the code begins to fill it again. This logic is also contained in subroutine TOUT.

Data saved is the same as in the previous version of EPIC3 with the exception that temperature is always zero for a HULL metal (MTYPE=6) and strains are saved for an element if the SAIL option STRAIN is greater than zero.

If the SAIL option CATPLT is greater than zero, the time history file will be automatically cataloged. Any previous file for the problem will be copied to a new file at restart time and the new file will be extended as new data is accumulated. It is up to the user to purge this old file after the new one is cataloged.

G. HULL LINK ROUTINES

Subroutine MOTION was changed to accomodate an input velocity or HULL input tape. If NHULL is greater than zero and IXYZ for the node is such that the node should be driven by velocity a call is made to subroutine HVEL to find node velocities UHULL, VHULL, and WHULL ($\dot{X}$, $\dot{Y}$, and $\dot{Z}$). If velocities cannot be found, the node is processed as a normal node and velocities are determined from applied forces. If velocities are found these are placed in appropriate node storage and the applied force calculation is avoided. If NVEL is greater than zero, IXYZ is checked and the input velocity is applied if appropriate.

Subroutine HVEL directs all HULL input tape activities. When it is first called, HVEL calls RDH to read the header blocks from the HULL input tape, Tape 14, and to fill in initial station positions for the NSTN HULL stations. In addition, the arrays UHULL, VHULL, WHULL, XHULL, YHULL and ZHULL are initialized. These arrays are doubly dimensioned. The first dimension corresponds to the HULL station number. The second dimension is 1 or 2 and is flip-flopped as new data is read into core. The flip-flop is required since, in general, the EPIC problem time will be between two HULL times at which data was saved. Linear interpolation will be employed to find HULL velocity values at the required EPIC time.

On subsequent calls, HVEL will replenish the HULL data arrays as EPIC time exceeds the time at which HULL data in core is valid.

HULL station position is interpolated between the two HULL times and this interpolated position is compared in HVEL to the position of the node requiring velocity data. Velocity data is acquired from a HULL station if:

a. the station contains the projectile material at both HULL times, and

b. it is the nearest station containing appropriate material to the node and is within a DELTA distance of the node (DELTA is set in a DATA statement in HVEL to 1 centimeter).

If a station cannot be assigned, X velocity is returned as -7777777 and the node velocity is determined from applied forces.

The user should be advised that if the SAIL option DEBUG is greater than zero a great quantity of data will output from this section of EPIC.

H. OTHER MAIN CHANGES

The HULL metal equation of state is described in another section of this report. It is implemented in subroutine PHULL.

Subroutine BREAK was changed to employ maximum principle strain (if STRAIN is greater than zero) and a P/Y failure model to determine if an element has failed. In addition, it was modified to provide a printout of pertinent element information for the first 100 failures. The ICHECK element word carries information concerning element failure. If ICHECK=0, the element has never failed. If ICHECK is greater than zero it has failed. If ICHECK is less than zero the maximum tensile strain has exceeded the HULL metal input EBS and is not subject to further calculations.

If the SAIL option FAIL is greater than zero and ICHECK is greater than zero, the element is not allowed to have a tensile pressure or any yield strength.

Subroutines STRAIN and ISTRESS were modified to calculate total strains from the incremental strain rates and the EPIC time increment if STRAIN is greater than zero.

The main routine was modified to interface with the EPIC Monitor program.

In addition to these changes, MAIN was modified extensively to incorporate SAIL updating features to reduce core size, reduce the number of "IF" statements, implement desired LCM options, etc.

MAIN INPUT SUMMARY

I. GENERAL CONTROL CARD

CASE, NCYCLE, IPRES, CPMAX, EMAX, DTMAX, DTMIN, SSF, TMAX, NHULL

II. LINKED RUN INPUT - IF NHULL WAS EVER SET TO 1

NMLINK, MATHUL(1), MATEPIC(1), ..., MATHUL(NMLINK), MATEPIC(NMLINK)
ZSTOP, VHIMP, THSTRT, NHELE, DTHULL
NHDROP, THDROP(1), NHEDROP(1), ..., THDROP(NHDROP), NHEDRP(NHDROP)

III. PRESSURE INPUT CARDS - IF IPRES=2

ELE1, ELEN, ELEINC, N1, NN, NODEINC, PRES
0
PTIME, P(T)
0

IV. TIME HISTORY PLOT CARDS

ISPLOT, ILPLOT, NNPLOT, NLPLOT, DTSYS

ONLY IF NPLOT(1)=0

NPLOT(1), X(1), Y(1), Z(1) . . . . . .

ONLY IF NPLOT(NNPLOT)=0

. . . NPLOT(NNPLOT) X(NNPLOT) Y(NNPLOT) Z(NNPLOT)

ONLY IF LPLOT(1)=0

LPLOT(1), X(1), Y(1), Z(1) . . . . . .

ONLY IF LPLOT(NLPLOT)=0

. . . LPLOT(NLPLOT), X)NLPLOT), Y(NLPLOT), Z(NLPLOT)

V. DATA OUTPUT CARDS

TWRITE, ECHECK, ILOAD, ISAVE

GENERAL CONTROL CARD

CASE is the floating point problem number.

NCYCLE is the desired cycle number for restart. A value of -1 means a restart from the latest cycle.

IPRES controls applied pressure input.
IPRES=0 means there will be no applied pressure cards.

IPRES=1 uses the applied pressure from the previous run.

IPRES=2 will read new applied pressure data.

CPMAX is the maximum CP time to be used for the problem. This is a redundant input now since EPIC uses CP time on the job card to terminate. Setting CPMAX to zero bypasses this feature.

EMAX — is the upper limit for total kinetic energy if applied pressures are included (IPRES=1 or 2). This input is used as a check for numerical instability and the code is stopped if kinetic energy exceeds EMAX. Unlike previous versions, EMAX is always input whether needed or not. If set to zero all checks will be bypassed.

DTMAX — is the maximum integration time increment.

DTMIN — is the minimum time increment allowed. If exceeded, the run will terminate.

SSF — is a multiplier for the time step calculation. It should be less than 1.

TMAX — is the maximum problem time for this run.

NHULL — is set to 1 if this is to be a linked HULL/EPIC run. Set it to 0 otherwise.

LINKED RUN INPUT

DTHULL — Minimum time step allowed for surface projectile elements prior to dropping them.

NHELE — Elements to be dropped through the use of DTHULL must be greater than this number.

THSTRT — Time to begin dropping NHELE elements if DTHULL criterion is met.

NMLINK — The number of EPIC projectile materials which can be linked to HULL input.

MATHUL(I)
MATEPIC(I) — These arrays define the relationships between HULL material numbers and EPIC material numbers.

ZSTOP — A linked run stops when minimum projectile Z reaches ZSTOP assuming an added Z value of VHIMP * TIME.

VHIMP — Velocity of projectile used to determine if projectile is at a Z value less than ZSTOP.

NHDROP — Number of (THDROP, NHEDRP) pairs.

THDROP(I)
NHEDRP(I) — Elements greater than or equal to element number NHEDRP(I) are dropped when TIME reaches THDROP(I).

PRESSURE INPUT CARDS

SPATIAL DESCRIPTION CARD(S)

ELE1	is the first element in a series to which pressure is applied. Must not be less than ELE1 or ELEN from a previous pressure card.
ELEN	is the last element in the series of elements. It cannot be less than ELE1.
ELENINC	is the increment between ELE1 and ELEN.
N1	is the node number opposite the triangular face of element ELE1 to which pressure is applied.
NN	is the node number opposite the triangular face of element ELEN.
NODEINC	is the node increment between N1 and NN.
PRES	is the pressure applied to the triangular faces of the elements described on the card.

The last card in this series should be a crad with a single 0 on it.

PRESSURE MULTIPLIER CARDS

PTIME	is the time corresponding to P(T).
P(T)	is the factor by which all pressures are multiplied at the corresponding time.

Cards must be input in order of increasing time.

The last card should be a card with a single 0 on it.

TIME HISTORY PLOT CARDS

GENERAL CARD

ISPLOT	is set to 1 if system time history data should be saved. It is zero otherwise.
ILPLOT	is set to 1 if internal loads time data should be saved. It is zero otherwise.
NNPLOT	is set to the number of nodes at which time data is desired.

DTSYS — is the time increment for saving system and internal loads data (if DTSYS is zero it is set to TMAX/25)

NODE PLOT CARD

NPLOT(I) — is the node number of the Ith node at which time history data is desired.

(X(I), Y(I), Z(I)) — is the spatial point at which node data is desired (not required unless NPLOT(I) is zero).

ELEMENT PLOT CARD

LPLOT(I) — is the element number of the Ith element at which time history data is desired.

(X(I), Y(I), Z(I)) — is the spatial point at which element data is desired (not required unless LPLOT(I) is zero).

DATA OUTPUT CARDS

TWRITE — is the time increment at which output will be provided (the problem time of the restart dump is added to this value to determine the time of the first data output).

ECHECK — is a code to determine the type of printed output.

If ECHECK is less than 900 system data, individual node data and data for elements with a plastic strain equal to or greater than ECHECK will be printed.

If ECHECK is 999, system data will be printed and nodal data on the plane of symmetry at Y=0. No element data is printed.

If ECHECK is greater than 1,000, only system data is printed.

ILOAD — is a print code for internal loads data. If ILOAD=0 no internal loads data is printed. If ILOAD=1 internal loads are printed for the cylindrical projectile based on values of NlL, NOR, NIR and NPL as defined in PREP input.

<u>ISAVE</u> is a flag to determine if a restart file should be made at this time for subsequent plotting.
If ISAVE=0, no file will be made.
If ISAVE=1, a file will be made.

There can be any number of these data output cards. Subroutine READIN will discover an end-of-file and not attempt to read additional cards. In this case, the last value of TWRITE will be used as a continuing time increment for data output.

SECTION VII
CREATING A DUMP FOR TRANSFER TO HULL

Program HULLIN and its sizing program HULLINSZ are used to create a Tape 15 which KEEL can use to transfer the EPIC projectile back into HULL. The user specifies a desired cycle for transfer along with desired materials and desired spatial regions. Program HULLIN reads the EPIC dump at the required cycle and creates a tape for use by KEEL. Data on the tape is as follows:

RECORD	DATA
1	CASE, NHELE, NHELER, STRAIN, XHMIN, XHMAX, YHMIN, YHMAX, ZHMIN, ZHMAX, (DESC(1), ..., DESC(8)), NW, NMATIN, MATIN(1), AMSS(1), AMOMX(1), AMOMY(1), AMOMZ(1), AIE(1), AKE(1), ..., MATIN(NMATIN), AMSS(NMATIN), AMOMX(NMATIN, AMOMY(NMATIN), AMOMZ(NMATIN), AIE(NMATIN), AKE (NMATIN)

where

CASE is the EPIC case number

NHELE is the number of elements to transfer to HULL

NHELER is the number of elements per record on Tape 15

STRAIN is the value of the SAIL variable

XHMIN, ..., ZHMAX are min and max coordinates for the projectile being transferred

DESC(1), ..., DESC(8) is the problem description from the dump tape

NW is the number of words per element being transferred (27 if STRAIN=0, 33 if STRAIN=1)

NMATIN is the number of separate materials being transferred

MATIN, AMSS, ..., AKE are values of material number, mass, XYZ linear momenta, total internal energy and total kinetic energy for all of the materials being transferred

2, ...	Records, each of which contains data for NHELER elements. The data for each element is:
	MATERIAL NUMBER
	X_1, Y_1, Z_1 X_2, Y_2, Z_2 X_3, Y_3, Z_3 X_4, Y_4, Z_4 } positions for the defining nodes
	U V W } Element centroid velocity values (i.e., one-fourth the sum of the values at the four node points)
	Mass of the element
	Density of the element
	Total internal energy of the element
	Pressure X and Y stress deviators XY, XZ, YZ shear stresses
	ICHECK XYZ normal strains XY, XZ, YZ shear strains } if STRAIN > 0
	$\bar{\varepsilon}_p$ - accumulated plastic strain
LAST	X, Y, Z values for all EPIC3 element time history points in the region being transferred to HULL

User input to HULLIN is seen below:

PROBLEM DESCRIPTION

CASE, CYCLE

MATERIAL TRANSFER DESCRIPTION

NMATIN, MATIN(1), ..., MATIN(NMATIN)

X, Y, Z LIMITS FOR TRANSFER

XMIN, XMAX, YMIN, YMAX, ZMIN, ZMAX

<u>NMATIN</u>	is the number of materials to transfer
<u>MATIN(I)</u>	is an array containing material numbers for transfer
<u>XMIN, ..., ZMAX</u>	define spatial regions for transfer

To facilitate an analysis of the adequacy of the transfer, program HULLIN prints mass, momenta and energies, for each respective material being transferred. These can be compared to similar data provided by KEEL as it processes Tape 15.

SECTION VIII
CHANGES TO THE STATE PLOTTER

The state plotter, POST2, was minimally changed to accomodate SAIL options, automatic restart tape attachment and cataloging of Tape 99, the new time history internal laods format and to provide room for future contour variable expansion. In addition, a sizing program, P1SIZE, was written to read SAIL options from restart (or time history) tapes, check LCM availability and provide an INPUT2 file for SAIL processing of POST1.

A few changes were made in input in addition to field free format. Case number must be read as the first input. This is required in order to attach the restart tape. Titles for plots have been dropped and the problem description from tape inserted at the top of each plot. Internal loads are now specified by a card beginning with a 100 (instead of 15). This change was made to leave room for future expansion of the contour plots variable array. Of course, the previously terminating blank card is now a card with a single zero on it.

Input for POST1 is as follows:

CASE NUMBER CARD

CASE

GEOMETRY PLOT CARDS AS REQUIRED

GEOMETRY PLOT CARD

1, VIEW, CYCLE IAXES, IFAIL

PLOT LIMIT CARD IF IAXES=1

XMAX, XMIN, YMAX, YMIN, ZMAX, ZMIN

3D PERSPECTIVE CARD IF VIEW=4

XEYE, YEYE, ZEYE, XPLANE, YPLANE, ZPLANE, LHIDE

VELOCITY VECTOR PLOT CARDS AS REQUIRED

PLOT CARD

2, VIEW, CYCLE, IAXES, EDGE, AROW, VSCALE

LIMIT CARD IF IAXES=1

XMAX, XMIN, YMAX, YMIN, ZMAX, ZMIN

3D PERSPECTIVE CARD FOR VIEW=4

XEYE, YEYE, ZEYE, XPLANE, YPLANE, ZPLANE

CONTOUR PLOT CARDS AS REQUIRED

PLOT CARD

TYPE, VIEW, CYCLE, IAXES, EDGE, NLINE, PRINT, ISYM

LIMIT CARD IF IAXES=1

XMAX, XMIN, YMAX, YMIN, ZMAX, ZMIN

CONTOUR SPECIFICATION CARD

VAR(1), VAR(2), ..., VAR(NLINE)

GEOMETRY, VELOCITY, CONTOUR PLOTS TERMINATING CARD

0

INTERNAL LOADS PLOT CARD(S) AS REQUIRED

PLOT CARD

100, CYCLE, IAXES, IP, IV, IM

LIMIT CARD FOR IAXES=1

PMAX, PMIN, VMAX, BMAX, BMIN, LENGTH

INTERNAL PLOTS TERMINATING CARD

0

VIEW	is a code specifying the view requested. VIEW=1,2,3 requests two-dimensional plots of the XZ, YZ and XY axes respectively VIEW=4 requests a three-dimensional plot
CYCLE	is the cycle number of the dump desired to be plotted
IAXES	determines if automatic scaling will prevail IAXES=0 means automatic scaling IAXES=1 means that coordinate limits will be supplied
IFAIL	places an "X" at the center of each failed element for VIEW=1,2 or 3; i.e., each element for which ICHECK is greater than zero. (ICHECK=-1 elements are considered as separated from the problem and never checked)
XMAX, XMIN, YMAX, YMIN, ZMAX, ZMIN	are coordinate limits for the plot and are used only if IAXES=1
IEYE, YEYE, ZEYE	are coordinates of the Observer
XPLANE, YPLANE, ZPLANE	are coordinates included in the plane for which results should be plotted. The plane is normal to a line from (XEYE, YEYE, ZEYE) to (XPLANE, YPLANE, ZPLANE)
LHIDE	specifies if lines should be hidden LHIDE=0 means plot all free surfaces LHIDE=1 means plot only those surfaces not hidden from the observer (It is recommended that this option be employed only with all nodes in core to avoid excessive CP time)

EDGE	specifies whether an outline should be drawn around the external surfaces for a velocity vector plot EDGE=1 means plot the outline EDGE=0 means do not plot
AROW	if set to 1 will draw arrow heads on the velocity vectors
VSCALE	is a multiplier for velocity components. If set at zero, VSCALE will give the longest vector a length of two percent of the vertical axis
TYPE	specifies the type of contour plot desired

TYPE	CONTOUR
3	Pressure
4-6	XYZ normal stresses
7	XY shear stress
8	XZ shear stress
9	YZ shear stress
10	Effective stress (VonMises limit if element is plastic)
11	Equivalent plastic strain
12	Temperature
13	Internal energy per unit volume
14	Plastic work per unit volume

NLINE	specifies the number of contours to be plotted. This number is currently limited to 8. If NLINE=0, a default option is invoked for 6 contours at values of 5, 20, 40, 60, 80 and 95 percent of the range between the minimum and maximum variable quantity limits.
PRINT	prints nodal quantities for the specified variable if set to 1. If the print is not desired set it to 0.
ISYM	specifies the increment at which symbols are placed on contour lines. ISYM=1 places symbols at the forward end of each contour line within an element. ISYM=5 places symbols at the forward end of each fifth element, etc. ISYM=0 will not put any symbols on the contour lines.

<u>VAR(I)</u>	specifies the magnitude of contours to be plotted if NLINE is greater than zero
<u>IP</u>	= 1 gives plots of axial loads
<u>IV</u>	= 1 gives plots of shear loads
<u>IM</u>	= 1 gives plots of bending moments
<u>PMAX, PMIN</u>	are bounding coordinates for the vertical axial load axis (force)
<u>VMAX, VMIN</u>	are bounding coordinates for the vertical shear load axis (force)
<u>BMAX, BMIN</u>	are bounding coordinates for the vertical bending moment axis (force-distance)
<u>LENGTH</u>	is the maximum coordinate of the horizontal axis which is defined as the deformed centerline distance from the free end of the rod (rod length)

SECTION IX
TIME HISTORY PLOTTING

The time history plotter, POST2, as well as the method of saving time history data were all but completely rewritten. Some POST2 rewrite was required because of the method of packing data on the time history file. However, a large amount of rewrite was required because of the old POST2 requirement that data for four types of time history stations had to fit in addressable core. This imposed severe restrictions on the amount of data which could be plotted during any single run. Titles in A6 format also had to be changed to provide reasonable labels on a 60 bit word size machine.

The approach taken to processing the data was to employ separate files for storage of element, node, system and internal loads data. In particular, when data for an element, a node, the system or a plane for internal loads is requested, POST2 reads the EPIC time history tape and places all data for that element, node, etc. on a file. This file is then read by the appropriate plotting subroutine and the desired variable (pressure, stress components, etc.) is plotted. Because of this processing technique, it is desirable to call for all required plots for a particular element node or plane prior to switching the request to a new element, node or plane.

The requirement to submit a plot title with each plot request has been eliminated. Instead, the problem description assigned in PREP will automatically be recovered from the time history tape and placed at the top of each plot.

The addition of strains to data carried in elements (if the SAIL option STRAIN is greater than zero) allowed us to change POST2 to offer time history plots of ε_{XX}, ε_{YY}, ε_{ZZ}, ε_{XY}, ε_{XZ}, ε_{YZ} and the maximum principle strain for elements. In addition, if PYFAIL is greater than zero in the problem calculation, P/Y can be plotted as a function of time for an element.

The P2SIZE program is run prior to running POST2. This program reads the SAIL option block from the time history tape and prepares an INPUT2 file for SAIL to use when processing POST2.

Similarly to PREP, MAIN and POST1, POST2 will automatically attach the latest time history file if CATPLT is greater than zero. In addition, it will automatically catalog the Tape 99 file if CATPLOT is greater than zero. The file is cataloged on the A machine at BRL under a directory specified by the value of the SAIL option USER. The file is cataloged as:

P2XXXXPTYYYY

where the problem (case) number is XXXX.YYYY. If the time history tape is a real tape it is recommended that it be copied to a disc in the control stream because of the many potential READ's and REWIND's which can occur in POST2.

Input for the new POST2 is as follows:

CASE NUMBER CARD

CASE

SYSTEM PLOT CARD(S) AS REQUIRED

TYPE, IAXES, SCALE, TMAX, TMIN, VMAX, VMIN

INTERNAL LOAD CARD(S) AS REQUIRED

TYPE, IAXES, LAYER, SCALE, TMAX, TMIN, VMAX, VMIN

NODE PLOT CARDS AS REQUIRED

TYPE, IAXES, NODE, SCALE, TMAX, TMIN, VMAX, VMIN

ELEMENT PLOT CARDS AS REQUIRED

TYPE, IAXES, ELE, SCALE, TMAX, TMIN, VMAX, VMIN

ENDING CARD

0

<u>TMAX, TMIN, VMAX, VMIN</u> these min and max values for each plot will be used if IAXES=1. If IAXES=0, min and max values are automatically chosen. Note that values for TMAX, TMIN, VMAX, and VMIN must be present on the card even if IAXES=0. In this case, any values - including zeros are acceptable.

<u>SCALE</u> is a multiplier on the variable specified by the input card

<u>NODE</u> is specific node number

<u>ELE</u> is specific element number

<u>LAYER</u> is specific layer or plane number for internal loads plots

<u>TYPE</u> specifies the variable to be plotted

The acceptable values are:

TYPE	MEANING	
1	SYSTEM	Total Energy
2	SYSTEM	Kinetic Energy
3	SYSTEM	Internal Energy
4	SYSTEM	Plastic Energy
5-7	SYSTEM	XYZ Maximum Coordinates
8-10	SYSTEM	XYZ Minimum Coordinates
11-13	SYSTEM	XYZ Centers of Gravity
14-16	SYSTEM	XYZ Linear Momenta
17-19	SYSTEM	XYZ Velocities
20	SYSTEM	Total Velocity
21-23	SYSTEM	XYZ Angular Momenta
24-26	SYSTEM	XYZ Angular Velocities
27	SYSTEM	Axial Load
28	SYSTEM	Shear Load
29	SYSTEM	Bending Load
30-32	NODE	XYZ Positions
33-35	NODE	XYZ Velocities
36-38	NODE	XYZ Accelerations
39	NODE	Node Pressure
40	ELEMENT	Pressure
41-43	ELEMENT	XYZ Normal Stresses
44	ELEMENT	XY Shear Stress
45	ELEMENT	XZ Shear Stress
46	ELEMENT	YZ Shear Stress
47	ELEMENT	Effective Stress (Von Mises limit if element in plastic flow)
48	ELEMENT	εp-equivalent plastic strain
49	ELEMENT	Temperature
50	ELEMENT	Internal Energy per unit volume
51	ELEMENT	Plastic work per unit volume
52-54	ELEMENT	XYZ Normal Strains
55	ELEMENT	XY Shear Strain
56	ELEMENT	XZ Shear Strain
57	ELEMENT	YZ Shear Strain
58	ELEMENT	Maximum Principle Strain
59	ELEMENT	P/Y

NOTE: These last two quantities are nodal averages if the MAIN run was made with IPNODE=1.

SECTION X
EPIC MONITOR SYSTEM

Routines were developed to allow the user to communicate with a running EPIC3 job for job status determination and job control. The routines were patterned after those in use in the HULL system. Communications is provided through a file catalogued as EPIC3CONTROL on sysset on the BRL Z machine. The file is catalogued under the directory KIMSEY.

Communications is currently limited to program MAIN, but it would be a simple matter to add the necessary routines to PREP or the plotting programs. Subroutine CONTW in MAIN (actually brought into MAIN from PROLOGUE) is called at the end of each cycle if the SAIL option RCONTRL is greater than zero. Subroutine CONTW checks that the file EPIC3CONTROL exists by trying to attach it. If the file does not exist, it creates one. The random access file is created using mass store commands - so that attaches and catalogs can be made without accompanying system printouts.

The file is currently configured for twenty-one fifty word records. The first record is a table containing case numbers for the following data records. For example, if the fifth word in the first record is 2.5, then the fifth data record (sixth record on the tape) contains the data for problem (case) 2.5. A zero in the first twenty words of the first record indicates an available data record.

Each data record for a running (or previously run) problem contains the following information:

WORD NUMBER	DATA
1	Current value of NCYCLE
2	Current value of TIME
3	Current value of DT
4	Controlling element, LCRIT
5-12	Problem description
13	Central Memory being used
14	LCM being used
15	CP time used during the current run
16	CP time per element per cycle for the run to this time
17	Date of last file update for the run
18	Time of day for last file update
19	The number of elements in the problem, NELE
20	The number of nodes in the problem, NNODE
21	The number of system data records written on the time history tape
22	The number of internal loads records

23	The number of node records
24	The number of element records
25	The value of LCMNODE
26	The value of LCMNBUF
27	The value of LCMELEM
28	The value of LCMEBUF
29	The number of node blocks in the problem, NNBLK
30	The number of element blocks in the problem, NEBLK
31	The current value of NBAND
32	The current value of LBAND
33	Not used
34	NHULL
35	MODE=EPIC or END depending on whether the problem has terminated or not
36-49	Not used
50	Control word=0 if no control has yet been exercised =1 if the problem should be terminated =2 if a printout of the output file to this point is desired (the control word is input from program EMON which the user must interactively run to monitor EPIC3 runs)

As can be seen, there is sufficient room for a great deal more monitoring information and/or control options. Subroutine CONTW and program EMON are sufficiently commented so that they could easily be modified by the BRL to provide more data or control as desired.

Program EMON is a separate program on the EPIC SYSTEM file. It can be separated from the file by running the following control cards on the A machine.

```
Job card
Account card
ATTACH, HULLIB, ID=KIMSEY.
LIBRARY, HULLIB.
ATTACH, OLD, EPICSYS, ID=KIMSEY.
COPYCR, INPUT, A.
REWIND, A.
DYTHUL, I=A.
CATALOG, SAIL, EMON, ID=XXX.
*EOR
SAIL PROGRAM EMON
```

As a result of the run, the source file for EMON will be catalogued as file EMON on sysset directory XXX. The file can then be attached from a terminal and compiled on the A machine using the STATIC option. It must be run with the HULLIB library attached and declared. The Interactive compile command to invoke STATIC is:

```
FTN, STATIC, I=EMON, B=RUN
```

where RUN is the user's desired name for the load and execute file (the default is LGO).

Program EMON connects INPUT (Tape 5) and OUTPUT (Tape 6) to the machine on which it is running. Thus running interactively the user inputs requests on a remote terminal and the program output is automatically printed on the terminal.

Commands and data displayed from EMON are as illustrated below:

COMMAND:LIST

EMON: EMON prints a list of EPIC problems on the EPIC3CONTROL file. For each problem, EMON prints status (MODE), cycle, problem time and time increment.

COMMAND:DISPLAY XXXX.YYYY

EMON: EMON prints the data in the problem data record for case XXXX.YYYY displaying problem description, cycle, time, time increment, NHULL, NELE, NNODE, etc.

COMMAND:DIRECT XXXX.YYYY PARTOUT

EMON: EMON places a 2 in the control word for case XXXX.YYYY and when EPIC3 calls CONTW it will use this control word to return the output file making it available to the user.

COMMAND:DIRECT XXXX.YYYY STOP

EMON: EMON places a 1 in the control word for case XXXX.YYYY and when EPIC3 calls CONTW it uses this control word to cause EPIC termination.

COMMAND:STOP

EMON: EMON ceases execution.

SECTION XI
SAMPLE LINKED PROBLEM

A sample linked HULL/EPIC problem was run to demonstrate proper operation of the linking algorithms. The calculation with a comparison to experimental results will be presented in a separate technical report. In this section we will primarily discuss the input decks used for the EPIC portion of the problem.

Figure 1 presents density contours from the HULL calculation of an L/D=10 penetrator impacting a thin armor plate at approximately 1 km/sec velocity. Figure 2 presents EPIC velocity vectors at various times in the EPIC portion of the calculation. Forward element planes in the EPIC calculation were dropped at 36.5 and 42 microseconds resulting in the shortened rod seen at 60 microseconds. The planes were dropped because they became too highly compressed and were essentially stopping the calculation.

The initial HULL and EPIC geometries are seen in Figure 3. Since the hemispherical nose cap of the rod is eroded away early in the HULL calculation it was not modelled at all in the EPIC calculation. This does not affect the accuracy of the calculation since HULL velocities used to drive EPIC will be inside the rod at the correct locations and since the cap area will have been removed from the penetrator prior to termination of the HULL calculation. There were 541 HULL stations in the rod nose area.

The rod modelled in EPIC has 840 nodes and 3132 elements. The EPIC calculation required 2370 CP seconds to reach a problem time of 200 microseconds in 1530 cycles. Overall, the EPIC running time on the CDC7600 at BRL can be computed to be 1.84 milliseconds per node per cycle (all of the nodes were contained in LCM for the calculation). This is slightly greater than the usual 1.4 milliseconds because of the extra station location calculations required because of the link.

Figures 4 and 5 present PREP and MAIN input decks for the calculation in CGS units. As seen in these figures the rod nose was located at 0.385 cm (since the nose cap was not included) and the rod terminated at a Z value of 7.7 cm. It has a constant radius of 0.385 cm. The staballoy was modelled using the HULL equation of state with a P/Y failure curve determined from notched tensile tests. The SAIL variable FAIL was set to zero for the calculation. Most of the MAIN deck consists of EPIC time history station location data (34 element stations in all). The deck specifies that a dump be made every 20 microseconds and that the run be terminated at 200 microseconds or at the time when the minimum Z value of the rod minus 1.E5 X TIME reaches -20.085 cm. This is the location of the second target plate. Elements numbered 3025 and greater (the forward rod plane) will be dropped at 36.5

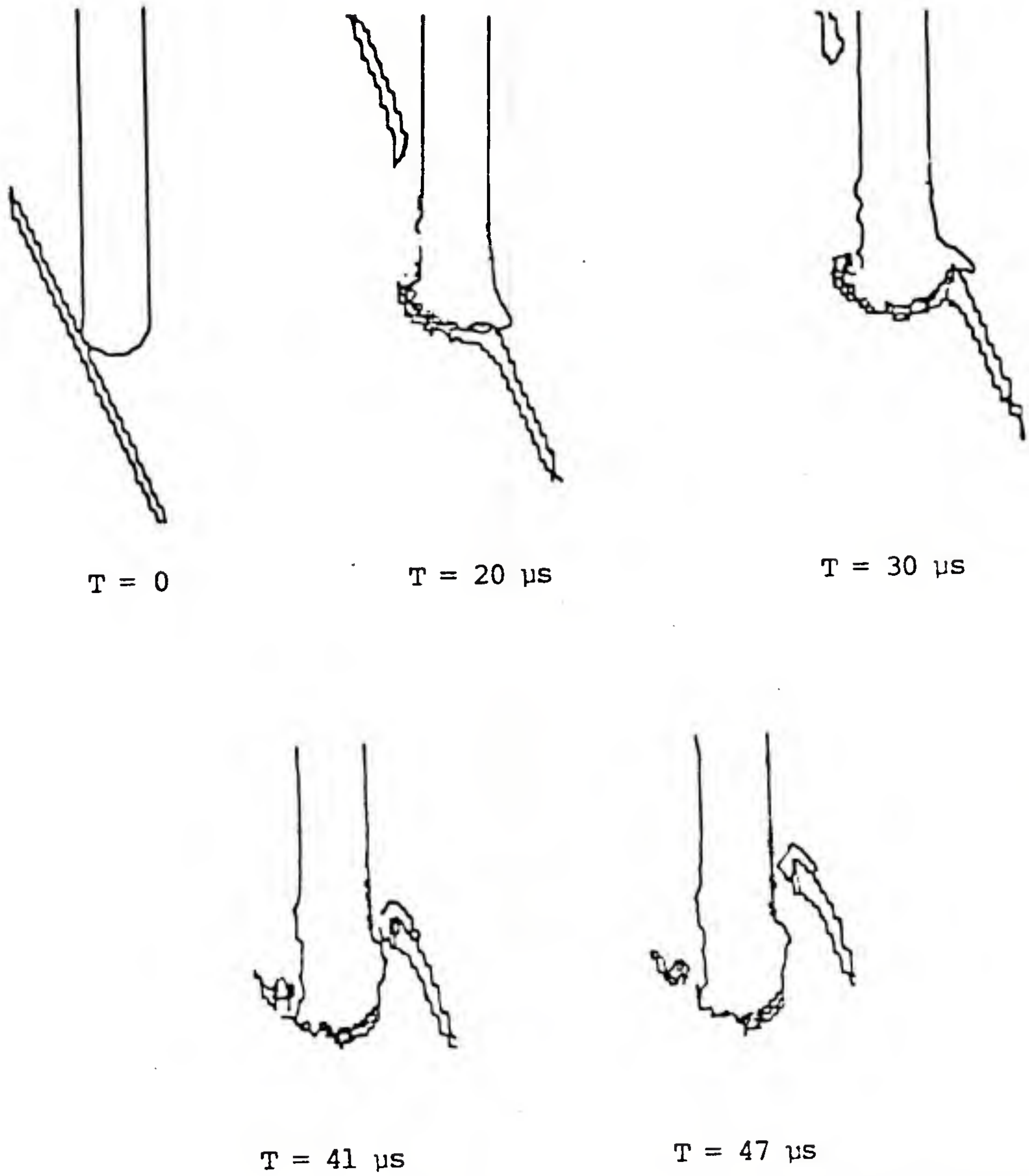


Figure 1. OTI/BRL HULL Calculation of RHA Perforated by staballoy at a velocity of 1 km/sec and an impact angle of 65 degrees.

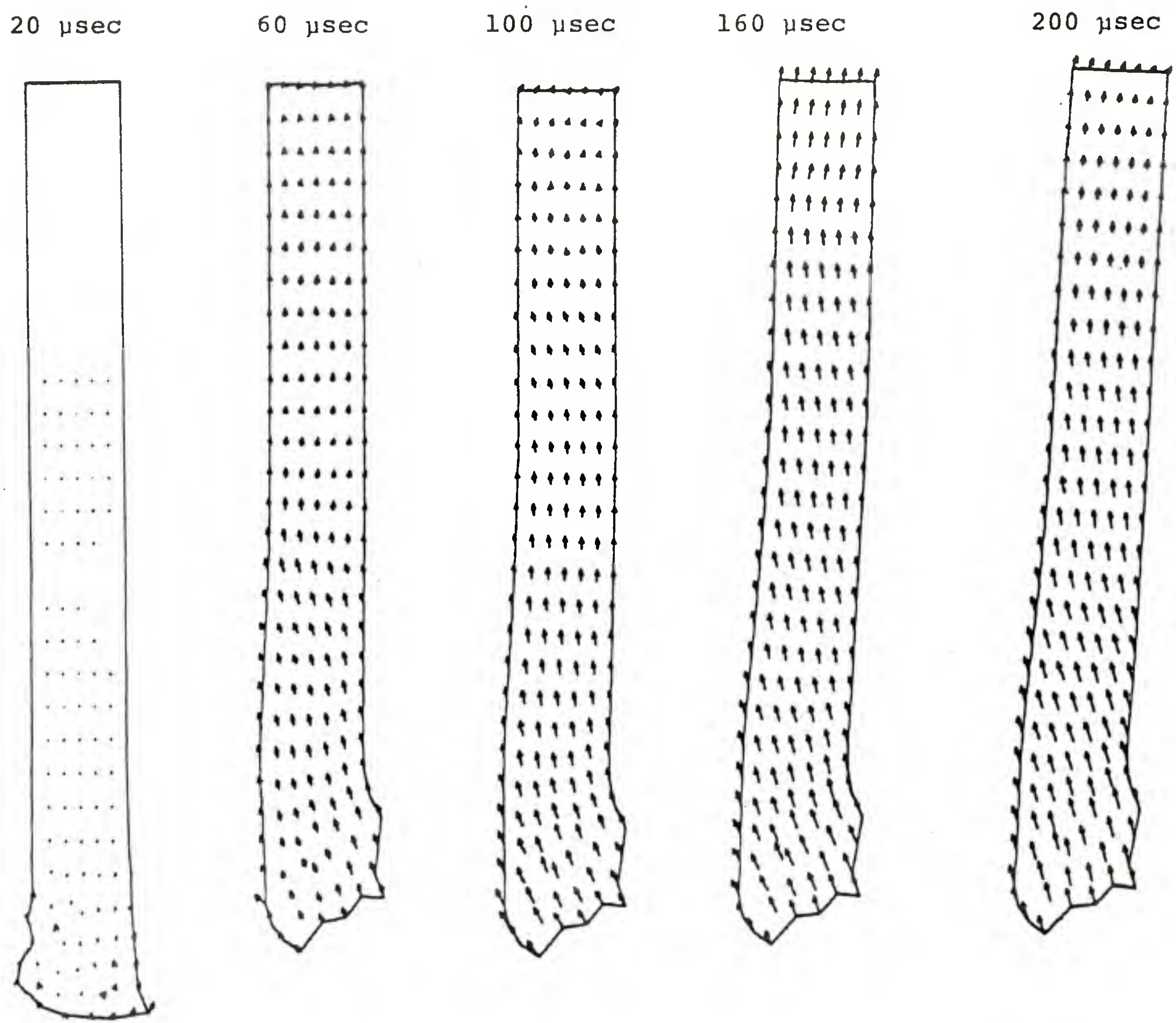


Figure 2. EPIC3 velocity Plots for Linked Calculation.

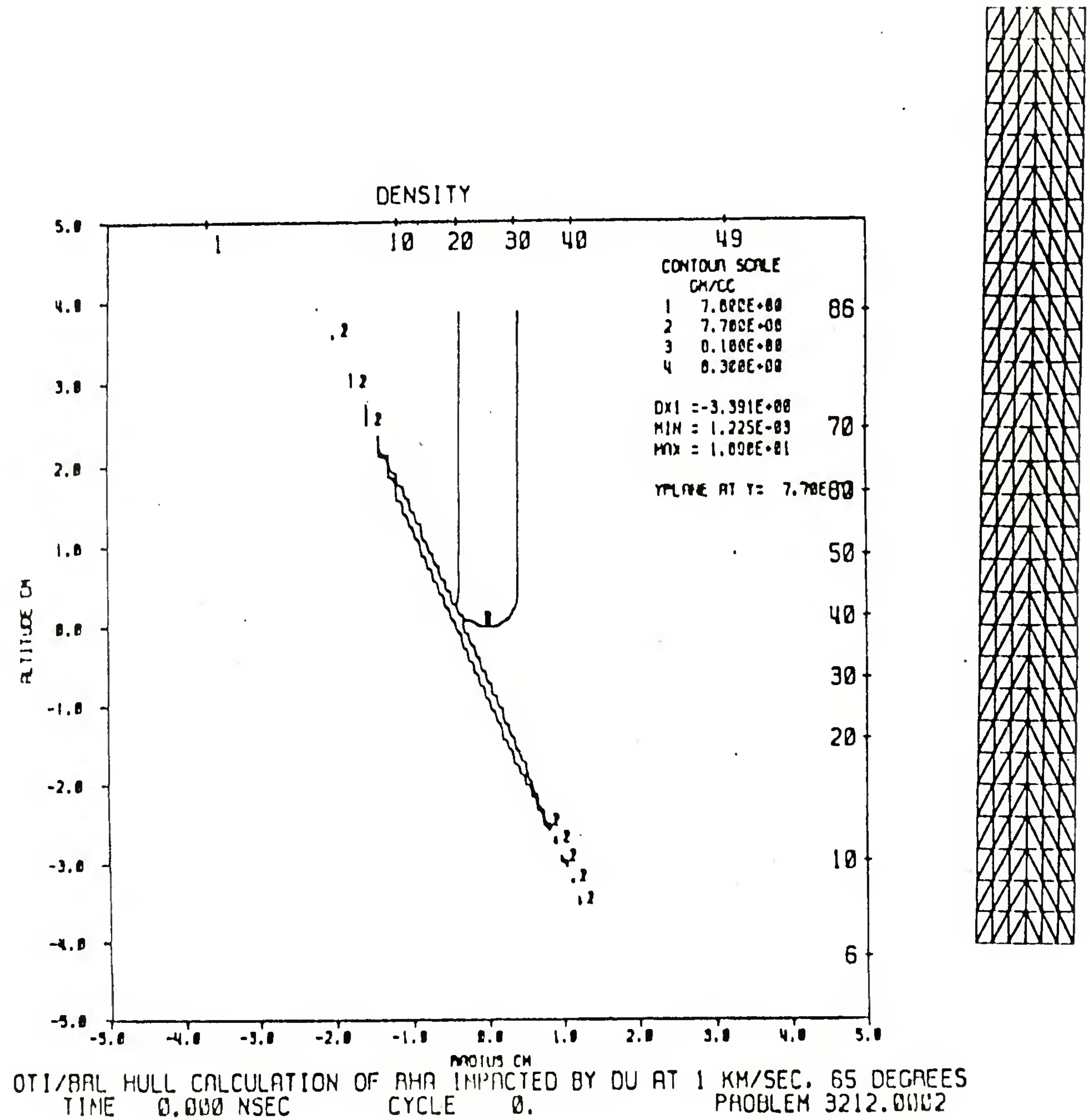


Figure 3. Initial HULL and EPIC Geometries

```
 OTI EPIC CALCULATION. L/D=10 STABALLOY ROD. LINKED HULL RUN
2.1 1 0 0 1 0 0 0 1 3 0 30
 1 6
 URANIUM 0.75 PERCENT TI
 18.9 2.48E5 1.53 1.6 -1.E20 0.3
 .2 4 10.E9 15.E9 .5 1.4E9 .49E9 .9 .5 .9 .5
 4 -0.9 .12 -0.6 .22 -0.333 0.36 0 2.5
 1.E100
 0
 1 1 1 0 0 0 0
 2
 3 0 30 0 7.7 .385 1 0 1
 .385 0 .385 0
 0
 0 0 0 0 0 0 0
 0
 2
 3 1 29 1 0 1
 0
 0
 0
 0 0 0
 0 0 0 0 0 0 1.E-8
```

Figure 4. PREP Input Data for the Sample Problem

```
2.1 -1 0 1.E20 1.E20 1 1.E-14 1 200.E-6 1
1 3 1
-20.085 -1.E5 1 10000 1 2
36.5E-6 3025
42.E-6 2917
1 1 0 34 5.E-6
0 0.05 0.05 .75
0 0.05 0.05 1.0
0 0.05 0.05 1.5
0 0.05 0.05 2.0
0 0.05 0.05 2.5
0 0.05 0.05 3.0
0 0.05 0.05 4.0
0 0.05 0.05 5.0
0 0.05 0.05 6.0
0 0.05 0.05 7.0
0 0.3 0.05 .4
0 0.3 0.05 1.0
0 0.3 0.05 2.0
0 0.3 0.05 3.0
0 0.3 0.05 4.0
0 0.3 0.05 5.0
0 0.3 0.05 6.0
0 0.3 0.05 7.0
0 -0.3 0.05 .4
0 -0.3 0.05 1.0
0 -0.3 0.05 2.0
0 -0.3 0.05 3.0
0 -0.3 0.05 4.0
0 -0.3 0.05 5.0
0 -0.3 0.05 6.0
0 -0.3 0.05 7.0
0 0.05 0.3 .4
0 0.05 0.3 1.0
0 0.05 0.3 2.0
0 0.05 0.3 3.0
0 0.05 0.3 4.0
0 0.05 0.3 5.0
0 0.05 0.3 6.0
0 0.05 0.3 7.0
20.E-6 1.E100 1 1
```

Figure 5. MAIN Input Data for the Sample Problem

microseconds. Elements numbered 2917 and greater (the second most forward plane) will be dropped at 42 microseconds. The automatic element drop routine which reacts to element time step was not used.

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314
1	Director Defense Advanced Research Projects Agency ATTN: Tech Info 1400 Wilson Boulevard Arlington, VA 22209
1	Director Defense Nuclear Agency Washington, DC 20305
1	Deputy Assistant Secretary of the Army (R&D) Department of the Army Washington, DC 20310
2	Commander US Army BMD Advanced Technology Center ATTN: BMDATC-M, Mr. P. Boyd Mr. S. Brockway PO Box 1500 Huntsville, AL 35807
1	HQDA (DAMA-ARP) WASH DC 20310
1	HQDA (DAMA-MS) WASH DC 20310
2	Commander US Army Engineer Waterways Experiment Station ATTN: Dr. P. Hadala Dr. B. Rohani PO Box 631 Vicksburg, MS 39180

No. of Copies	Organization
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
10	Commander US Army armament Research and Development Command ATTN: DRDAR-TD, Dr. R. Weigle DRDAR-LC, Dr. J. Frasier DRDAR-SC, Dr. D. Gyorog DRDAR-LCF, G. Demitrack DRDAR-LCA, G. Randers-Pehrson DRDAR-SCS-M, R. Kwatnoski DRDAR-LCU, E. Barrieres DRDAR-SCM, Dr. E. Bloore DRDAR-TSS (2 cys) Dover, NJ 07801
2	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Dr. Joseph E. Flaherty Watervliet, NY 12189
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120

DISTRIBUTION LIST

No. of Copies	Organization
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Communications Research and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
1	Commander US Army Missile Command ATTN: DRSMI-R Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-RBL Redstone Arsenal, AL 35809
1	Commander US Army Missile Command ATTN: DRSMI-YDL Redstone Arsenal, AL 35809
2	Commander US Army Tank Automotive Research & Development Cmd ATTN: DRDTA-UL V. H. Pagano Warren, MI 48090
1	Commander TARADCOM Tank Automotive Systems Laboratory ATTN: T. Dean Warren, MI 48090
6	Director US Army Materials and Mechanics Research Center ATTN: DRXMR-T, Mr. J. Bluhm Mr. J. Mescall Dr. M. Lenoe R. Shea F. Quigley DRXMR-ATL Watertown, MA 02172
2	Commander US Army Research Office ATTN: Dr. E. Saibel Dr. G. Mayer PO Box 12211 Research Triangle Park NC 27709
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL (Tech Lib) Whie Sands Missile Range NM 88002
1	Office of Naval Research Department of the Navy ATTN: Code ONR 439, N. Perrone 800 North Quincy Street Arlington, VA 22217

DISTRIBUTION LIST

No. of Copies	Organization
3	Commander Naval Air Systems Command ATTN: AIR-604 Washington, DC 20360
3	Commander Naval Ordnance Systems Command Washington, DC 20360
2	Commander Naval Air Development Center, Johnsville Warminster, PA 18974
1	Commander Naval Missile Center Point Mugu, CA 93041
2	Commander Naval Ship Engineering Center ATTN: J. Schell Tech Lib Washington, DC 20362
1	Commander and Director David W. Taylor Naval Ship Research and Development Center Bethesda, MD 20084
2	Commander Naval Surface Weapons Center ATTN: Dr. W. G. Soper Mr. N. Rupert Dahlgren, VA 22448
4	Commander Naval Surface Weapons Center ATTN: Dr. S. Fishman (2 cys) Code U-11, J. R. Renzi R. S. Gross Silver Spring, MD 20910
6	Commander Naval Surface Weapons Center ATTN: Code R-13, F.J. Zerilli K. Kim E. T. Toton M. J. Frankel Code K-22, F. Stecher J. M. Etheridge Silver Spring, MD 20910
3	Commander Naval Weapons Center ATTN: Code 31804, Mr. M. Smith Code 326, Mr. P. Cordle Code 3261, Mr. T. Zulkoski China Lake, CA 93555
6	Commander Naval Weapons Center ATTN: Code 3181, John Morrow Code 3261, Mr. C. Johnson Code 3171, Mr. B. Galloway Code 3831, Mr. M. Backman Mr. R. E. VanDevender,Jr. Dr. O.E.R. Heimdahl China Lake, CA 93555
2	Director Naval Research Laboratory ATTN: Dr. C. Sanday Dr. H. Pusey Washington, DC 20375
2	Superintendent Naval Postgraduate School ATTN: Dir of Lib Dr. R. Ball Monterey, CA 93940

DISTRIBUTION LIST

No. of Copies	Organization
1	Long Beach Naval Shipyard ATTN: R. Kessler T. Eto R. Fernandez Long Beach, CA 90822
1	HQ USAF/SAMI Washington, DC 20330
1	AFIS/INOT Washington, DC 20330
10	ADTC/DLJW (MAJ G. Spitale) Eglin AFB, FL 32542
20	ADTC/DLYV (Mr. J. Collins) Eglin AFB, FL 32542
1	AFATL/DLYV Eglin AFB, FL 32542
1	AFATL/DLODL Eglin AFB, FL 32542
1	AFATL/CC Eglin AFB, FL 32542
1	AFATL/DLODR Eglin AFB, FL 32542
1	HQ PACAF/DOOQ Hickam AFB, HI 96853
1	HQ PACAF/OA Hickam AFB, HI 96853

No. of Copies	Organization
1	OOALC/MMWMC Hill AFB, UT 84406
1	HQ TAC/DRA Langley AFB, VA 23665
1	TAC/INAT Langley AFB, VA 23665
1	AUL-LSE 71-249 Maxwell AFB, AL 36112
1	AFWAL/MLLN (Mr. T. Nicholas) Wright-Patterson AFB, OH 45433
1	ASD/ENESS (S. Johns) Wright-Patterson AFB, OH 45433
1	ASD/ENFEA Wright-Patterson AFB, OH 45433
1	ASD/XRP Wright-Patterson AFB, OH 45433
1	HQ USAFE/DOQ APO New York 09012
1	COMIPAC/I-32 Box 38 Camp H. I. Smith, HI 96861
10	Battelle Northwest Laboratories ATTN: G. D. Marr P. O. Box 999 Richland, WA 99352
1	Director Lawrence Livermore Laboratory ATTN: Ms. C. Westmoreland P. O. Box 808 Livermore, CA 94550

DISTRIBUTION LIST

No. of Copies	Organization
2	Lawrence Livermore Laboratory PO Box 808 ATTN: Dr. R. Werne Dr. J.O. Hallquist Livermore, CA 94550
6	Los Alamos Scientific Laboratory PO Box 1663 ATTN: Dr. R. Karpp Dr. J. Dienes Dr. J. Taylor Dr. E. Fugelso Dr. D. E. Upham Dr. R. Keyser Los Alamos, NM 87545
5	Sandia Laboratories ATTN: Dr. R. Woodfin Dr. M. Sears Dr. W. Herrmann Dr. L. Bertholf Dr. A. Chabai Albuquerque, NM 87115
1	Headquarters National Aeronautics and Space Administration Washington, DC 20546
1	Jet Propulsion Laboratory 4800 Oak Grove Drive ATTN: Dr. Ralph Chen Pasadena, CA 91102
1	Director National Aeronautics and Space Administration Langley Research Center Langley Station Hampton, VA 23365
1	US Geological Survey 2255 N. Gemini Drive ATTN: Dr. D. Roddy Flagstaff, AZ 86001

No. of Copies	Organization
1	Aerojet Ordnance Company 9236 East Hall Road Downey, CA 90241
1	Aeronautical Research Associates of Princeton, Inc. 50 Washington Road Princeton, NJ 08540
1	Aerospace Corporation 2350 E. El Segundo Blvd. ATTN: Mr. L. Rubin El Segundo, CA 90009
1	AVCO Systems Division 201 Lowell Street ATTN: Dr. Reinecke Wilmington, MA 01803
4	Battelle Columbus Laboratories 505 King Avenue ATTN: Dr. M. F. Kanninen Dr. G. T. Hahn Dr. L. E. Hulbert Dr. S. Sampath Columbus, OH 43201
3	Boeing Aerospace Company ATTN: Mr. R. G. Blaisdell (M.S. 40-25) Dr. N. A. Armstrong, C. J. Artura (M.S. 8C-23) Seattle, WA 98124
2	Brunswick Corporation 4300 Industrial Avenue ATTN: P. S. Chang R. Grover Lincoln, NE 68504
1	Computer Code Consultants, Inc. 1680 Camino Redondo ATTN: Dr. Wally Johnson Los Alamos, NM 87544

DISTRIBUTION LIST

No. of Copies	Organization
1	Dresser Center PO Box 1407 ATTN: Dr. M.S. Chawla Houston, TX 77001
1	Effects Technology, Inc. 5383 Hollister Avenue Santa Barbara, CA 93111
2	Firestone Defense Research and Products 1200 Firestone Parkway ATTN: R. L. Woodall L. E. Vescelius Akron, OH 44317
1	FMC Corporation Ordnance Engineering Division San Jose, CA 95114
1	Ford Aerospace and Communications Corporation Ford Road, PO Box A ATTN: L. K. Goodwin Newport Beach, CA 92660
1	General Atomic Company PO Box 81608 ATTN: R. M. Sullivan F. H. Ho S. Kwei San Diego, CA 92138
1	General Dynamics PO Box 2507 ATTN: J. H. Cuadros Pomona, CA 91745
1	General Electric Company Armament Systems Department Burlington, VT 05401
1	President General Research Corporation ATTN: Lib McLean, VA 22101

No. of Copies	Organization
1	Goodyear Aerospace Corporation 1210 Massilon Road Akron, OH 44315
1	H. P. White Laboratory 3114 Scarboro Road Street, MD 21154
4	Honeywell, Inc. Government and Aerospace Products Division ATTN: Mr. J. Blackburn Dr. G. Johnson Mr. R. Simpson Mr. K. H. Doeringsfeld 600 Second Street, NE Hopkins, MN 55343
1	Hughes Aircraft Corporation ATTN: Mr. W. Keppel MS M-5, Bldg. 808 Tucson, AZ 85706
1	International Applied Physics, Inc. ATTN: Mr. H. F. Swift 7546 McEwen Road Centerville, OH 45459
2	Kaman Sciences Corporation 1500 Garden of the Gods Road ATTN: Dr. P. Snow Dr. D. Williams Colorado Springs, CO 80933
1	Lockheed Palo Alto Research Laboratory 3251 Hanover Street ATTN: Org 5230, Bldg. 201 Mr. R. Robertson Palo Alto, CA 94394
1	Materials Research Laboratory, Inc. 1 Science Road Glenwood, IL 60427

DISTRIBUTION LIST

No. of Copies	Organization
2	McDonnell-Douglas Astronautics Company 5301 Bolsa Avenue ATTN: Dr. L. B. Greszczuk Dr. J. Wall Huntington Beach, CA 92647
1	New Mexico Institute of Mining and Technology ATTN: TERA Group Socorro, NM 87801
1	Northrup Corporation 3901 W. Broadway ATTN: R. L. Ramkumar Hawthorne, CA 90250
1	Nuclear Assurance Corporation 24 Executive Park West ATTN: T. C. Thompson Atlanta, GA 30245
1	Pacific Technical Corporation 460 Ward Drive ATTN: Dr. F. K. Feldmann Santa Barbara, CA 93105
2	Schumberger Well Services Perforating Center ATTN: J. E. Brooks J. Brookman PO Box A Rosharon, TX 77543
1	Science Applications, Inc. 101 Continental Boulevard Suite 310 El Segundo, CA 90245
1	Ship Systems, Inc. 11750 Sorrento Valley Road ATTN: Dr. G. G. Erickson San Diego, CA 92121

No. of Copies	Organization
1	Systems, Science and Software PO Box 1620 ATTN: Dr. R. Sedgwick La Jolla, CA 92038
2	TRW One Space Park, R1/2120 ATTN: D. Ausherman M. Bronstein Redondo Beach, CA 90277
1	United Technologies Research Center 438 Weir Street ATTN: P. R. Fitzpatrick Glastonbury, CT 06033
1	US Steel Corporation Research Center 125 Jamison Lane Monroeville, PA 15146
1	VPI & SU 106C Norris Hall ATTN: Dr. M. P. Kamat Blacksburg, VA 24061
2	Vought Corporation PO Box 225907 ATTN: Dr. G. Hough Dr. Paul M. Kenner Dallas, TX 75265
1	Westinghouse, Inc. PO Box 79 ATTN: J. Y. Fan W. Mifflin, PA 15122
1	Drexel University Department of Mechanical Engr. ATTN: Dr. P. C. Chou 32d and Chestnut Streets Philadelphia, PA 19104

DISTRIBUTION LIST

No. of Copies	Organization
1	Forrestal Research Center Aeronautical Eng. Laboratory Princeton University ATTN: Dr. A. Eringen Princeton, NJ 08540
3	Southwest Research Institute Dept. of Mechanical Sciences ATTN: Dr. U. Lindholm Dr. W. Baker Dr. R. White 8500 Culebra Road San Antonio, TX 78228
4	SRI International 333 Ravenswood Avenue ATTN: Dr. L. Seaman Dr. L. Curran Dr. D. Shockey Dr. A. L. Florence Menlo Park, CA 94025
1	State University of New York at Stony Brook Department of Materials Science and Engineering ATTN: Dr. H. Herman Stony Brook, NY 11790
2	University of Arizona Civil Engineering Department ATTN: Dr. D. A. DaDeppo Dr. R. Richard Tucson, AZ 85721
1	University of Arizona School of Engineering ATTN: Dean R. Gallagher Tucson, AZ 85721
1	University of California Los Angeles ATTN: Dr. M. Ziv Los Angeles, CA 90024
1	University of California Department of Physics ATTN: Dr. Harold Lewis Santa Barbara, CA 93106
2	University of California College of Engineering ATTN: Prof. W. Goldsmith Dr. A. G. Evans Berkeley, CA 94720
2	University of Delaware Department of Mechanical Engineering ATTN: Prof. J. Vinson Prof. B. Pipes Newark, DE 19711
1	University of Denver Denver Research Institute ATTN: Mr. R. F. Recht 2390 S. University Blvd. Denver, CO 80210
2	University of Florida Department of Engineering Sciences ATTN: Dr. R. L. Sierakowski Dr. L. E. Malvern Gainesville, FL 32601
1	University of Oklahoma School of Aerospace, Mechanical and Nuclear Engineering ATTN: Dr. C. W. Bert Norman, OK 73019
1	University of Vermont 201 Votey Bldg. ATTN: C. Brown Burlington, VT 05405

DISTRIBUTION LIST

No. of Copies	Organization	No. of Copies	Organization
	Aberdeen Proving Ground		
	Dir, USAMSAA ATTN: DRXSY-D DRXSY-MP, H. Cohen		
	Cdr, USATECOM ATTN: DRSTE-TO-F		
	Dir, USAAPG ATTN: Mr. S. Keithely, MTD		
	Dir, USACSL, EA ATTN: DRDAR-CLB-PA Bldg. E3516		

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number ______________________

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

____

____

____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) ______________________

____

____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

____

____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) ______________________

____

____

____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: ______________________

Telephone Number: ______________________

Organization Address: ______________________

____

____